AFRL-IF-RS-TR-2001-113
Final Technical Report
June 2001

# GUARD ARCHITECTURE FOR APPLICATION PORTABILITY

Fuentez Systems Concepts, Inc.

James A. Schmid

*APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.*

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2001-113 has been reviewed and is approved for publication.

APPROVED: CJ Maciag

CHESTER J. MACIAG
Project Engineer

FOR THE DIRECTOR:

WARREN H. DEBANY, Technical Advisor
Information Grid Division
Information Directorate

If your address has changed or if you wish to be removed from the Air Force Research Laboratory Rome Research Site mailing list, or if the addressee is no longer employed by your organization, please notify AFRL/IFGB, 525 Brooks Road, Rome, NY 13441-4505. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

| | |
|---|---|
| **REPORT DOCUMENTATION PAGE** | *Form Approved*<br>*OMB No. 0704-0188* |

| | | |
|---|---|---|
| **1. AGENCY USE ONLY** *(Leave blank)* | **2. REPORT DATE**<br>June 2001 | **3. REPORT TYPE AND DATES COVERED**<br>Final Jan 93 - Nov 96 |

**4. TITLE AND SUBTITLE**

GUARD ARCHITECTURE FOR APPLICATION PORTABILITY

**5. FUNDING NUMBERS**

C - F30602-93-0119
PE - 33140F
PR - 7820
TA - 04
WU - 12

**6. AUTHOR(S)**

James A. Schmid

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**

Fuentez Systems Concepts, Inc.
11781 Lee Jackson Highway, Suite 700
Fairfax Virginia 22203-3309

**8. PERFORMING ORGANIZATION REPORT NUMBER**

N/A

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

Air Force Research Laboratory/IFGB
525 Brooks Road
Rome New York 13441-4505

**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

AFRL-IF-RS-TR-2001-113

**11. SUPPLEMENTARY NOTES**

Air Force Research Laboratory Project Engineer: Chester J. Maciag/IFGB/(315) 330-3184

**12a. DISTRIBUTION AVAILABILITY STATEMENT**

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

**12b. DISTRIBUTION CODE**

**13. ABSTRACT** *(Maximum 200 words)*

This report describes the requirements, design considerations, and approach used to develop a highly-configurable computer network guard system. The report describes the composition of its two main parts: The Guard Configuration Utility (GCU), and the composed Guard Systems themselves. A discussion of the considerations for multi-platform design and development is included, followed by the final testing results and recommendations for future enhancement and usage.

**14. SUBJECT TERMS**

MLS, Multi-Level Security, Trusted Operating System, Platform Portability, Guard, Firewall

**15. NUMBER OF PAGES**

64

**16. PRICE CODE**

| | | | |
|---|---|---|---|
| **17. SECURITY CLASSIFICATION OF REPORT** | **18. SECURITY CLASSIFICATION OF THIS PAGE** | **19. SECURITY CLASSIFICATION OF ABSTRACT** | **20. LIMITATION OF ABSTRACT** |
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | UL |

# Table of Contents

# Figures

# Tables

# 1. Introduction

This Final Technical Report applies to the Guard Architecture for Application Portability (GAAP) system developed by Fuentez Systems Concepts, Inc. for the United States Air Force (USAF), Rome Laboratory.

## 1.1 Report Overview

The purpose of this document is to reiterate the reasons that lead to the concept of GAAP, restate the initial objectives of the GAAP system, describe the approach used to solve the identified problem, and draw conclusions and recommendations on how the GAAP system should be used and enhanced in the future. This report is being written for a general audience and includes descriptive graphics and tables to enhance understanding. As much as possible, project specific terms and extensive use of acronyms were avoided. A glossary is included to define any project terms used in this report and words that appear in the glossary are initially italicized for easy recognition.

The introduction section is used to define the initial problems that lead to the GAAP project concept and outlines the proposed solution. The System Objectives section describes how the initial system requirements for GAAP were defined. The Survey of MLS systems outlines the approach taken to select the *multi-level platforms* used for development. There is a section describing the prototype developed and how that aided in further understanding of the system. The GAAP Architecture section is a detailed discussion of operation of the GAAP system. The Final Testing Results are presented followed by Recommendations for future enhancement and usage.

## 1.2 Problem Definition

There are multiple situations that require the dissemination of information from one platform to another platform through network connections anywhere from a few feet to miles apart. It is often the case that one of these platforms contains information at a *level* above the recipient system. In these cases there must be assurances that the higher level information is never accessible by the lower level system. In these situations the most common solution to the problem is to implement a guard that either resides on the higher level system or exists as a separate platform between the two systems.

Using a guard as a means of protection of information has been effective as demonstrated by the sheer number of guard systems that have been developed. The problem is that for each solution, the final outcome of the development is a system that only runs on the target platform performing the functions defined for that particular environment. Typically the communications and filtering applications are specific to the operational environment having no flexibility for modification. This results in guards not just being an effective solution, but also a costly one.

## 1.3 Proposed Solution

It is from these observations that the concept of a Guard Architecture for Application Portability (GAAP) were formed. It was recognized that most guards share common functionality such as auditing, network configuration, administration, and message processing. The predominate reason an *accredited* operational guard is not suitable for new guard situations is that it is tied to a specific platform with no ability for modification to the message processing architecture.

From these observations, it was proposed that a guard system be constructed that provides those basic features such as auditing, administration, and control, but with specific objectives of platform portability to allow transition from one environment to another. It was also determined that the architecture should be as configurable as possible in the way information is processed and released. It should allow for new communications, interpretation, and filtering applications to be easily added and deleted to the guard system.

A concept was then formed defining two distinct systems. The first would be a configuration utility that would be used to maintain developed modules and provide an interactive ability to define the operational environment of a guard system. This utility would allow selection from a repertoire of guard modules and specification of the ordering of those selected modules. A final ability would be to transfer the selected information to a tape as a target guard system. Figure 1.3-1 shows how modules loaded into the *GAAP Configuration Utility* (GCU) are used to create a *Target Guard System* (TGS).

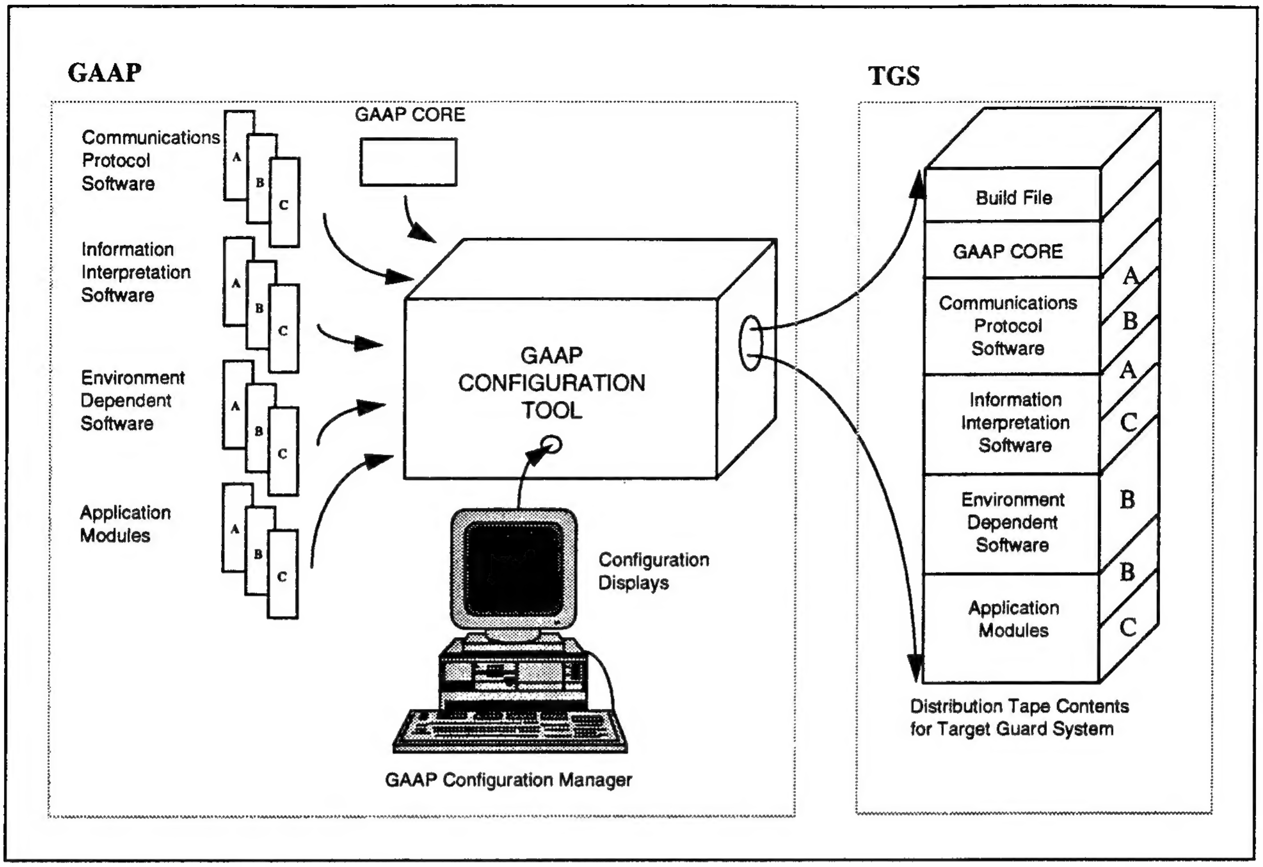

GAAP
TGS
GAAP CORE
Communications Protocol Software
A
B
C
Information Interpretation Software
A
B
C
Environment Dependent Software
A
B
C
Application Modules
A
B
C
GAAP CONFIGURATION TOOL
Configuration Displays
GAAP Configuration Manager
Build File
GAAP CORE
Communications Protocol Software
A
B
Information Interpretation Software
A
C
Environment Dependent Software
B
Application Modules
B
C
Distribution Tape Contents for Target Guard System

**Figure 1.3-1 The GAAP Configuration Utility**

The second system would be the guard software. The guard software would provide for administration and auditing of the guard along with a set of controlling software. These capabilities would be part of a core set of software included in all constructed guards. Also part of the guard system would be a set of application modules to include at a minimum input and output communications, format interpretation, validation, and filtering.

In mid 1993 when the project was just getting under way, multi-level system (MLS) platforms were emerging as a new direction for guards of the future. The benefit of the MLS platforms is separation of data with assurance as the selected MLS platforms are being evaluated by NSA. It was seen that the guard accreditation process could be greatly reduced by implementing guard specific software on an MLS platform. It was decided that a survey should be conducted to choose two MLS platforms for the GAAP development.

## 2. Survey of MLS systems

Identification of MLS systems to be considered for use in the GAAP development was obtained from listings in the NSA Evaluated Products List. This document provided a brief summary of each MLS system and the current status in the Trusted Computing System Evaluation Criteria (TCSEC) evaluation process. Information was obtained on each platform being considered and in most cases both the vendor and the NSA TCSEC group were contacted to get the very latest evaluation status. Other information obtained was the platform and operating system cost, support of an X windowing environment, networking capabilities, and POSIX compliance.

Further information on the platforms along with hands on demonstrations were witnessed at the National Information Systems Security Conference. Initially is was recommended that the DEC and the Harris Nighthawk be the two development platforms. In the interest of completing the development effort with a greater opportunity for transition to a target guard environment, the Sun Trusted Solaris was added to the selections list. The matrix shown in Table 2-1 shows the results of the MLS platform evaluation research.

| Vendor | AT&T | DEC | Harris | HP | HFSI | IBM | SecureWare | Sun Microsystems | Trusted Info System |
|---|---|---|---|---|---|---|---|---|---|
| Product | System V MLS | Ultrix MLS+ | CX/SX | HP-UX BLS | XTS-200 | AIX CMW | CMW+Ver1 | Solaris CMW | Trusted XENIX |
| Security Level | B1 | B1 | B1 | B1 | B3 | B1 | B1 | B1 | B2 |
| Operating System | UNIX | CMW | UNIX | UNIX | UNIX | CMW | CMW A/UX | CMW 1.0 | UNIX |
| Platform | 3B2, 386-486 | Vaxstation | Harris | HP 9000 | DPS 6000 | RISC 6000, 220-5xx | SCO UNIX / MAC | SPARC 10 | IBM PS/2, AT |
| Evaluation | Complete d RAMP | Formal | Formal 9/93 | Formal 9/93 | Completed | DAP | Completed | DAP | Completed |
| Evaluation Date | 9/1/92 | 6/1/89 | 12/1/91 | 8/11/92 | 5/27/92 | 6/1/91 | 1/30/91 | 6/1/89 | 4/8/92 |
| X Support | No | X11/R4 | X11/R4 | X11/R4 | No | X11/R4 | X11/R3 | X11/R4 | No |
| Trusted X | No | Yes | Yes | Yes | No | No, Vendor | TBD | Yes | No |
| Network | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Trusted Network | No | Yes | Yes | Eval. In Process | No | TBD | Yes | Yes | No |
| POSIX | No | 1003.1 | 1003.1<br>1003.2 | 1003.1<br>1003.2 | Future | 1003.1 | Yes | 1003.1 | No |

**Table 2-1 MLS Platform Evaluation Results**

## 3. System Requirements

This section defines the process in which the system requirements defined in the GAAP System Segment Specification were formulated. The GAAP Configuration Utility requirements were formulated from the proposed system concepts discussed in section 1.3. The concept of the GCU was further defined into two distinct parts. There would be a maintenance function that would allow for adding new *modules* to the configuration utility, maintenance of those modules, backup of the GCU in the event of a disk crash, and a review of audit information specific to the GCU system. The second part of the GCU would allow defining the hardware information for the guard environment defining the connected nodes and platform information. It would also allow for software selection from the maintained modules and specification of the order in which those modules execute in the target environment. A summary of the high level GCU system requirements are shown in Table 3-1 below.

| Functional Area | Description of Services |
|---|---|
| GCU Maintenance | Load new application modules |
| | Modification of module information |
| | Maintain loaded modules |
| | Review and reduction analysis of the audit trail |
| | Archive, restore and deletion of audit logs |
| | Provide means to backup the GCU |
| GCU Configuration | Selection of nodes in the guard environment |
| | Define connections between nodes |
| | Define of hardware platform information |
| | Selection of application module software |
| | Define the ordering of module software |
| | Maintenance of a defined configuration |
| | Assembly into a usable target guard system |
| | Transfer to a selected media device |

**Table 3-1 GAAP Configuration Utility Services**

Definition of the *GAAP Guard Software* (GGS) system requirements was more involved in that numerous functions exist in guards implemented to date and the goal was to select the functions most common to all guards without exceeding the scope of the effort. Mitre Corporation provided a comprehensive list of functions performed by ten different operational guard systems. This list was expanded to address the functions in terms of a portable and configurable architecture. In a Technical Interchange Meeting held at Rome Laboratory the list was pared down to the final system requirements for the GGS. A short discussion of the capabilities selected follows.

There are two distinct areas of functionality for the GGS. The first is the *core* software that provides functions that would be required by all guard systems. These include administration, auditing, monitoring, and controlling of messages passed through the guard. The second are the guard application modules that include communications, interpretation, and filtering.

Core administration requirements were identified to maintain the configured hosts, the configured communications applications, allow for setting the rejection counts, configuration of the port levels as well a enabling and disabling hosts and ports. A need was also identified to set the audit space threshold and change state if this threshold was exceeded. A system monitor was defined to aid in showing the current state of the guard and some history of how the messages were processed in the system. In support of the audit capability, there would be interactive displays for audit review, reduction, and setting of auditing criteria. Non-interactive software would include the ability to audit defined events and a system control used to define message processing, maintain the state of the system, and provide notifications.

Given that the application modules are specific to a guard environment and the GAAP system not being developed for a specific environment, there was less emphasis on elaborate application modules in favor of a strong framework for using those modules. Application modules were identified to provide receipt of data using FTP and mail, provide CRC validation, a filter based on word searches, allow for review of rejected messages, and release messages using FTP or mail. A summary of GGS system requirements is shown in Table 2-2.

| Functional Area | Description of Services |
|---|---|
| Alarms | Configuration of the events that trigger alarms |
| | Selection of the type of alarms such as audible or visual |
| Audit | Operational assurance of the audit mechanism |
| | Review and reduction analysis of the audit trail |
| | Archive, restore and deletion of audit logs |
| System Administration | Provide capabilities to define the network parameters |
| | Allow setting the level of a port |
| | Allow for clearing of rejection counts |
| | Provide the status of the system on a monitor screen |
| | Maintain the state of the system |
| | Check on the health of the system |
| Receipt of Data | Receipt of data using FTP to get files from a remote system |
| | Receipt of data on the guard from FTP or mail |
| | Provide a checksum validation |
| | Interpret data and perform source and destination checks |
| Data Filtering | Provide a capability to identify words from a configured list |
| | Allow for review of rejected messages |
| | Provide a forward on authority capability |
| Release of Data | Provide a downgrade for release of valid messages |

| | |
|---|---|
| | Release valid messages via FTP and mail |

**Table 2-2 Guard Software Services**

From the survey of functions commonly performed by guards there were a number of capabilities that are required by the GAAP system that are satisfied by use of a multi-level operating system. These functions are shown in Table 2-3.

| Functional Area | Description of Services |
|---|---|
| Labeling | Provide color coded screen labeling |
| | Mark hardcopy output with security labels |
| | Allow import of unlabeled data |
| | Write security label to magnetic media |
| | Read security label from magnetic media |
| | Provide security labels to all screen data |

**Table 2-3 Services provided by the operating system**

## 4. Prototype Demonstration

Once the software requirements were defined in the GAAP SRS for the GCU and GGS, a prototype was constructed to ensure an understanding of the implementation of the software requirements. This was accomplished by doing a rapid prototype constructed over a 4 week period. Data files were constructed to get a realistic view of the system. The prototype was demonstrated at the Technical Interchange meeting in February of 1994.

The prototype development brought to light significant issues in how the network configuration would be accomplished. Namely that the input communications module configuration would be tied to the defined hosts rather than to a specific format type. Improvements to the audit review and reduction capabilities were also identified. Ideas on how modules would be maintained and a concept of a graphical configuration utility were demonstrated.

Other advantages to development of this prototype included forcing configuration and setup of the hardware far in advance of unit code development. It also served to aid in training of X/Motif window code development. Overall, the development of the prototype after requirements analysis gave a vision of the system about a year and a half before delivery and proved to be very worth the effort required.

## 5. GAAP System Architecture

This section will define the GAAP system architecture to show how the system requirements were implemented. It will first define security considerations that applied to development of both the GCU and the GGS. There will then be a detailed discussion of the implementation of the two CSCIs. Following the detailed architecture of the GAAP System there will be a section discussing all the areas of the system that are configurable with the current implementation.

Both the GCU and GGS software rely on a library of common software. The common software provides functions for performing operations used in many places and simplifies operations that are required for modules to operate properly. Functions provided include building messages, getting unique id numbers, printing functions, archive/restore functions, path building functions and error reporting functions.

Multi-Level table functions allow data to be stored in tables at the data level. Processes can read data that is at or below its level. The common functions make the data appear to be in a single table.

GIDS access functions are used in the guard and provide a single point of access to the GIDS data base. Functions to read, write, update, and delete GIDS records are provided. Functions are also provided to access the message files associated with a data product. The GIDS access functions provided a simplified interface for applications to access the GIDS DB.

Both the GCU and GGS rely on a library of EDS software. The EDS software allows for platform portability by isolating operating system functions to a linked library of platform specific calls. The libraries in the EDS include Table Services, Process Services, System Services, Security Services, and Timer Services. The EDS functions provide a uniform interface to operating system provided call regardless of the platform.

### 5.1 Security Considerations

All security-related functionality used by the GAAP systems will be handled by the operating systems when possible. The Trusted Computer System Evaluation Criteria (TCSEC) Class B1 evaluated operating systems were selected in order that they could provide capabilities not available in untrusted operating systems. The architecture and design of both the GAAP Configuration Utility and the GAAP Guard Software have been developed so that these capabilities can be taken advantage of in an optimum manner.

All security-related functionality used by the GAAP systems will be handled by the operating systems. Only trusted processes, Audit, Control and Security break security policy. Communication between processes is via system provided IPC queues. Audit, control and security utilize a queue for each level of process running in the system. The untrusted processes read and write to queues only at their process level. Data is stored in multiple single level tables, with data stored only in a table at its level. Again, untrusted processes write only to tables that are at their process level.

## 5.2 GCU CSCI Architecture

The major components of the GCU are the GAAP Maintenance Software (GMS) and the Configuration Utility (CU) software. Other components include the audit interface, system control, security, and the audit process. These components and their relationship to each other are shown in Figure 5.2-1.

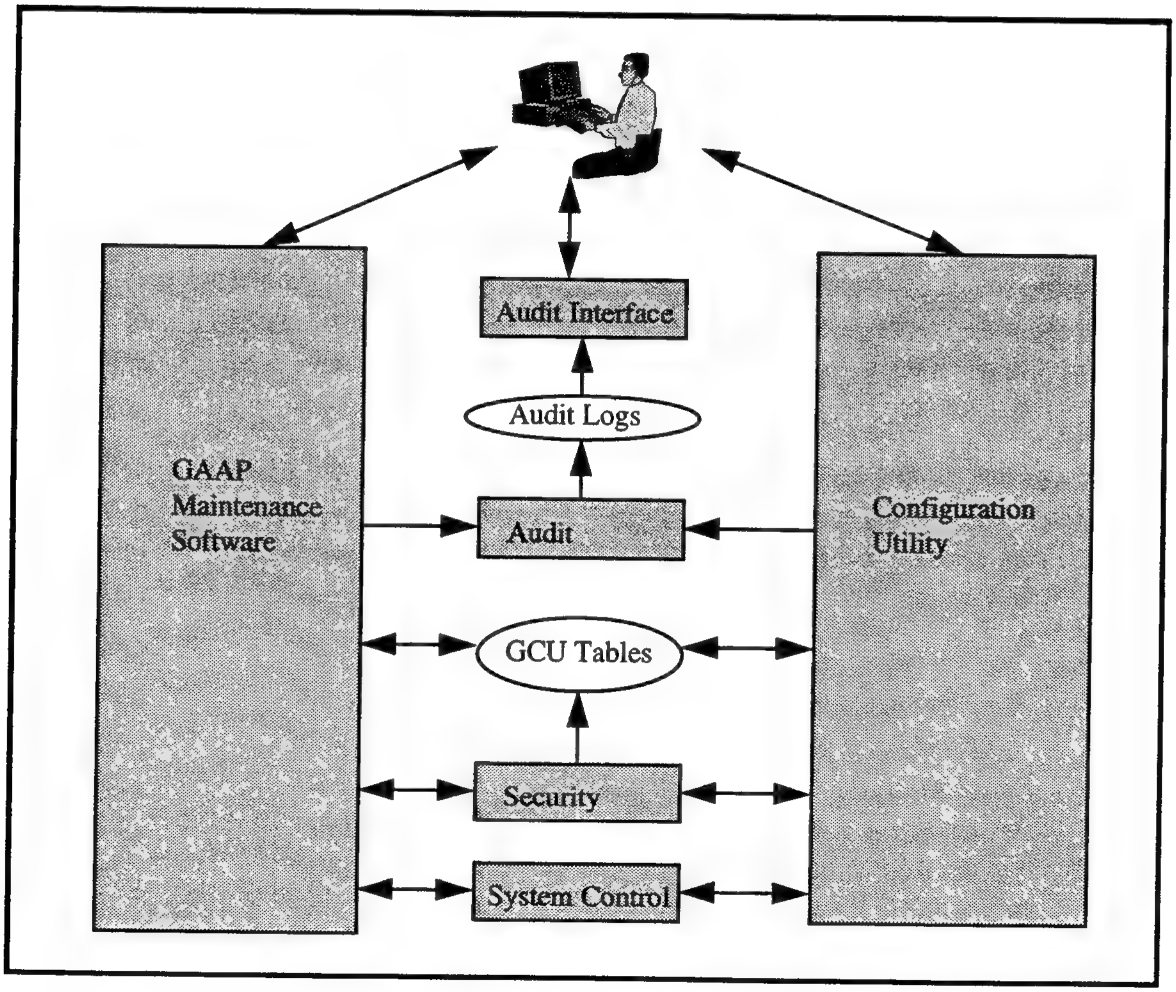


**Figure 5.2-1 GCU Interfaces**

There are user interfaces between the GMS, the CU, and the audit interface components. The GMS and CU both write to the audit process. The audit interface process will read the audit logs during operator review. Both the GMS and the CU write to tables to store information about the modules loaded and the stored configurations. These tables will be identified in later sections. The security process also writes to these tables as required by the security policy. System control maintains the running processes and performs health check operations while the system is running. Detailed information of the architecture and implementation of the GMS and CU are presented in the paragraphs that follow.

### 5.2.1 GAAP Maintenance Software (GMS)

The GMS is a process that allows for maintenance of the application modules developed for a guard. It provides the capability to load new modules and maintain summary information on those modules. Modules are loaded into defined groups that are also maintained by the operator. The GMS has a facility to allow backup and restore of the entire GCU in the case of a fatal disk crash. The functions performed by the GMS, including displays of some of the major interfacing windows follow.

#### 5.2.1.1 GMS Module Maintenance

The maintenance software allows for loading, deleting, manipulating, and reviewing of application modules. Functions performed by the GMS Maintenance Software follow.

##### 5.2.1.1.1 Class and Group Structure for Maintaining Modules

Module maintenance contains the functions for loading and maintaining modules used in the Configuration Utility. In order to allow for operator ease in identifying and classifying loaded modules, a scheme was developed where there are high level *classes* under which modules are loaded. Table 5.2.1.1-1 shows the defined classes with a description of the types of modules identified with each class.

| Class Name | Description |
|---|---|
| Communications | Communications modules for receipt and release |
| Communications Validation | Validation routines such as a CRC check |
| Information Interpretation | Message translation software |
| Information Interpretation Rulesets | Defined formats supported by the translation software |
| Library Applications | Other applications such as filtering and review |
| Core | Core software to be included in all guards |
| Unsupported | Software to be included with a guard distribution but not to be run as part of that guard. |

**Table 5.2.1.1-1 GMS Class Descriptions**

The module classes are not modifiable. To allow the operator to further distinguish the modules, a concept of *groups* of modules within classes was conceived. The operator can create and delete any number of groups within the defined classes for use in loading modules. A separate window and set of functions were developed to aid in the maintenance of these groups as described in 5.2.1.2 GMS Group Maintenance.

#### 5.2.1.1.2 Loading Modules

The tape that contains new modules to be loaded must contain a header file that contains the names of each module on the tape and a position relative to the start position of one. The GMS contains functionality to display a list of available *transport devices* and allows the operator to pick the device to read from. There is an option to read the header file off of the tape and display a list of the modules on the device read.

The operator chooses the class and group where the module will be loaded and starts the transfer. While the module is being transferred the status of reading the module is displayed. Once the module has been read onto the GCU disk, the summary information file is parsed to get the module information. If there is an error reading the summary file the module is removed and an error is displayed. If the module is loaded correctly, it is ready for review.

#### 5.2.1.1.3 Reviewing Modules

The main display used for review of loaded modules is shown in Figure 5.2.1.1.3-1. The layout shows the classes, groups and modules within each group. Each module is displayed with its classification and only modules at or below the level of the logged in operator are displayed. For each module displayed, summary information can be viewed and modified. Summary information available for each module is defined in Table 5.2.1.1.3-1.

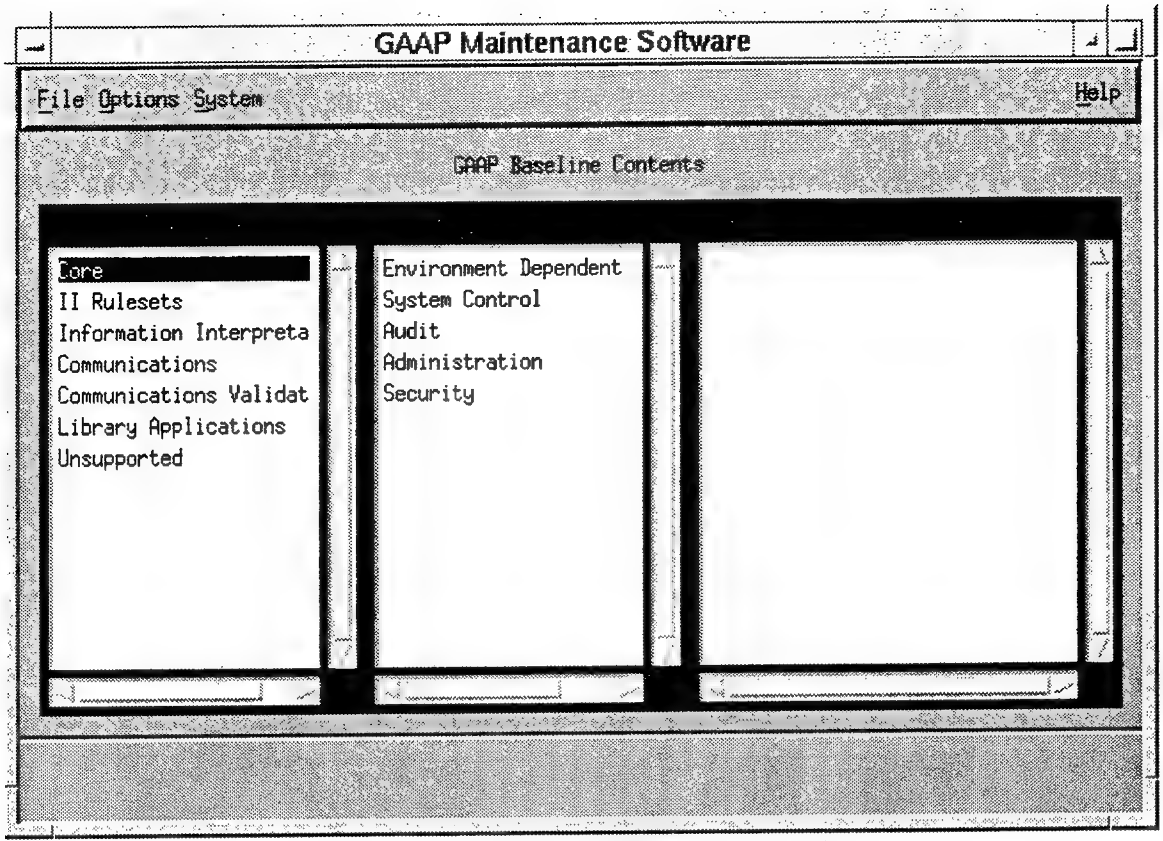


**Figure 5.2.1.1.3-1 GMS Review Display**

| Field Name | Field Description |
|---|---|
| Module Name | Displayable name of the module |
| Version | Version number |
| Vendor | Vendor Name |
| Point of Contact | A point of contact for the developed module |
| Phone Number | Phone number of the point of contact |
| Date Installed | Date the module was installed |
| Uncompiled Size | The uncompiled size of the module |
| Compiled Size | The compiled size of the module |
| Application Type | The type of application, eg. Communications input, Communications Output, Application |
| Executable Name | Name of the process executable |
| Processing Mode | Process to be permanent, transient, or unconfigured |
| Allowable States | States where the module is allowed to execute |
| Multi-Level Application | Requires multi-level queues to be created |
| Security Queue Required | Requires communications with security process |
| Error Handler | Provides error handling capability |
| Configuration Application | Name of an associated communications application |
| Application Authorization | Authorization required for the module |

**Table 5.2.1.1.3-1 Summary Information Fields**

When a new class is selected, the groups within that class are displayed. When a new group within a class is selected, the modules in that group are displayed. This allows for easy identification of modules using the class and group combination. The class and group combination is used in the GMS for the group maintenance and also for regrouping of modules as described in the Group Maintenance section.

Another feature of the GMS is allowing modules to be *certified* as tested on a selected platform. Through the summary information window there is an option that opens a certification display. This display allows the operator to mark a selected module as being certified on any platforms that have been configured through the Configuration Utility. This window allows entry of ancillary text to explain the circumstances of the certification. It is at the operators discretion to decide what qualifies as being certified, but the intention during the implementation was that the module had been fully tested on a platform and is ready to be included in new guards to be generated. It was not intended to mean that all certified modules had been evaluated or accredited on the basis of security.

Through options available to the module maintenance window, selected modules may be deleted one at a time. It is only possible to delete a module if the operator is logged in at system high.

### 5.2.1.2 GMS Group Maintenance

The GMS provides for maintenance of groups defined within the static classes. New groups can be added by the operator through the group maintenance window. Groups may only be deleted if they are empty requiring module deletions for every module in the group. The operator is not required to be at the same level of the group to delete the group as no level specific information is associated with a group.

Other group operations include regroup in which a module is moved from one existing group to another existing group. This is done in a similar window showing duplicate sets of classes, groups, and modules and selecting a source and destination for the module move.

### 5.2.1.3 GMS Backup

The GMS contains a feature to backup the entire installed GCU to include all modules, module information, code, configurations, etc. that exist under the installed directory. This is done to provide a backup in the case the disk becomes corrupt. There is a feature for auto remind to alert the operator to perform a backup after a specified period of time. Naturally there is the reciprocal restore operation to recover the information if necessary.

### 5.2.2 Configuration Utility (CU)

The Configuration Utility is a process that allows for definition and generation of the Target Guard System. It provides the capability to configure the system layout, hardware characteristics, and selected software for a new guard system. The ordering of the selected modules is defined along with roles and error configuration information. The functions performed by the CU, including displays of some of the major interfacing windows follow.

#### 5.2.2.1 Hardware Selection

The main window for the Configuration Utility is show in Figure 5.2.2.1-1. This figure shows a constructed system layout for a defined configuration. The first step in defining a new configuration is to ensure the platforms to be used have been configured. Platform information includes the operating system, the amount of disk space available, the roles for the platform, the port names and types, and selection of a loaded EDS layer for the platform. Platform information can be saved under an operator specified name and there is a save as function for instances where an old platform is close enough to be modified.

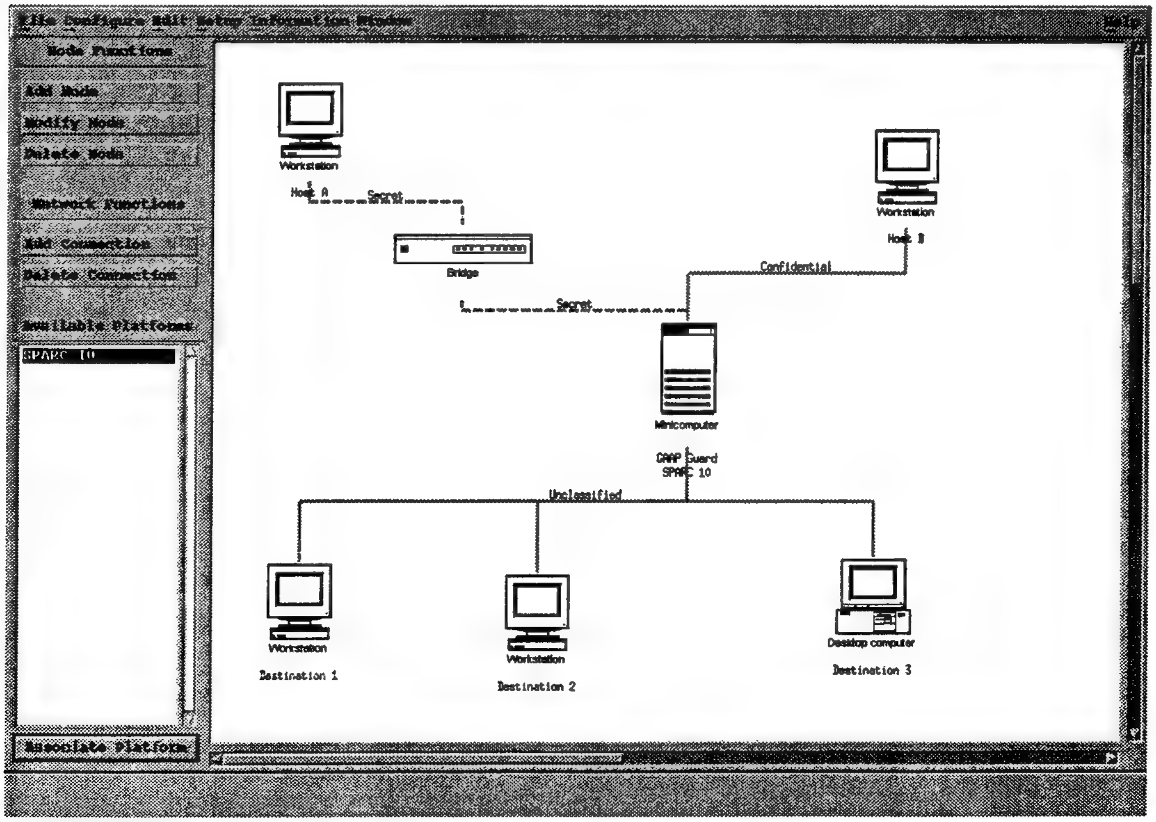


**Figure 5.2.2.1-1 Configuration Utility System Layout**

The next step in defining a configuration is to define the *nodes* that make up the guard environment. These typically would include the High and Low hosts and a guard host. When the node is added, a node type is associated with it. Node types include workstation, PC, and router. They also are defined as either a host node or a guard node. Once a node has been added to the layout it may be manipulated by dragging it around on the screen. It can then be associated with a platform configured in the first step.

Configurations can be saved along with specific information about the configuration for later access. As a configuration is being built, information is written to temporary tables to maintain data in the event of a crash. Explicit save operations are not required at every step as information is being written to temporary tables as windows are opened and closed.

### 5.2.2.2 Software Selection

Once the configuration has been defined, information specific to selecting the software and how that software will execute in the guard environment can be defined. The first step in software selection is to select a node from the configuration. The options are then available to select the software that will be associated with that node.
A list is displayed showing classes, groups, and modules. Similarly to the GMS, the operator picks the class and group to get a list of available modules. Modules are displayed as certified or uncertified. If a module is not certified it can still be added to a node after confirmation to a warning that uncertified modules are not guaranteed to run on the selected platform.

The next steps will define the operations of the Target Guard System that will be installed. Through the role configuration window the operator defined roles and the authorizations of those roles that will exist in the new guard. Typical roles of a guard are the system administrator and system operator. Other roles are specific to the Compartemented Mode Workstations (CMW) such as Information System Security Officer (ISSO) and System Administrator (SA). These roles have specific authorizations that define what an operator logged in is allowed and not allowed to do. Authorizations specific to the TGS are shown in Table 5.2.2.2-1. Some of these authorizations are only available by selecting a higher level authorization. For example, selecting gaap_admin causes the system administration window to appear on login. It also enables the choices of starting network configuration and audit review from the system administration main window. Use of authorizations is discussed in the GAAP Trusted Facility Manual.

| Authorization | Description of Authorization |
|---|---|
| gaap_monitor | Allows display of the guard monitor |
| gaap_admin | Allows display of the system administration window |
| gaap_background | Allows running of  processes in the background |
| gaap_netconfig | Allows option to run network configuration |
| gaap_auditcriteria | Allows running the audit criteria window |

| | |
|---|---|
| gaap_auditreview | Allows running audit review |
| gaap_auditarchiverestore | Allows performing archive and restore operations |
| gaap_statemaint | Provides the ability to the operator to perform state changes |
| gaap_appconfig | Allows the operator to start new applications |
| gaap_diskcheck | Allows the operator to setup the disk space check |
| gaap_dirty_word_search | Allows the operator to run dirty word search configuration |
| gaap_rejection_review | Allows the operator to run rejection review |

**Table 5.2.2.2-1 Guard Specific Authorizations**

A set of typically defined TGS roles with appropriate authorizations is shown in Table 5.2.2.2-2.

| | |
|---|---|
| Gaap Operator | gaap_admin |
| | gaap_appconfig |
| | gaap_dirty_word_search |
| | gaap_rejection_review |
| Gaap Administrator | gaap_admin |
| | gaap_netconfig |
| | gaap_auditreview |
| | gaap_auditarchiverestore |
| | gaap_auditcriteria |
| | gaap_statemaint |
| | gaap_diskcheck |
| | gaap_monitor |
| | gaap_background |
| | gaap_netconfig |
| Gaap Monitor | gaap_monitor |

**Table 5.2.2.2-2 Typical Guard Roles with Associated Authorizations**

After role configuration, the error handling for the TGS is defined. A window is displayed that defines errors that would occur using the set of application modules developed for the GAAP system. There are three actions the user has to choose from. The first is to send error notification to the sender of the message if an error occurs. The second action is reject for rejection review. In this case, a rejected message will be available to rejection review when this is selected. The final action is for messages that have been successfully released to send a confirmation of delivery to the sending host. Notification and rejection review selections are not available if confirmation is selected.

The final step in defining the Target Guard System is to select the *message formats* that will be processed and define the order in which modules will execute for those selected formats. A window is displayed showing the names of recognizable rulesets used to identify parts of a message. These are then used to define all the parts of the message format to be supported. There are a maximum of ten parts that may be assigned to a configured message format. The initial GAAP System supports the *USAFE Tum header*, the *USAFE mail header*, the *NITF header*, and a newly defined *GAAP header*. Combinations of these can be used to define the message expected to be processed. As an example Table 5.2.2.2-3 shows the definition of an NITF passed through the guard with a GAAP header used to identify source, destination, and classification information followed by the NITF message body. The final eight allowable parts are left undefined.

| |
|---|
| GAAP Header ruleset |
| NITF ruleset |
| Undefined |
| Undefined |
| Undefined |
| Undefined |
| Undefined |
| Undefined |
| Undefined |
| Undefined |

**Table 5.2.2.2-3 Parts to a Format Definition**

These formats are then usable in the module ordering window. In addition to the rulesets the operator may also fill spots with an "Any" category. This category means there will be some header or message text in that spot but the translation will not fail if it can not be identified.

To define the module ordering, the first step is to define the distinct paths through the system by defining a row for each type of data. One additional row is needed for specifying the interpretation software. Since the format is not known until it has been interpreted, there must be a format identified as "Any" used to execute the first module. This module will be the interpretation module. A row will exist for each additional format defined. The operator picks from an available list of software to order the execution of modules.

Figure 5.2.2.2-1 shows an example module ordering for a guard system that processes two type of messages. One type of message is an NITF and the other is a USAFE mail message. The Any format is used to allow the interpretation software to determine the type of message to be processed. Once it has been identified as an NITF, the security module downgrades the module and it is released using destination information contained in the header. If a USAFE mail message is identified, the message is passed through the dirty word search module followed by downgrade by the security module before being released. The module ordering creates the Data Product Routing Table used by the TGS system control process to route messages through the defined guard.

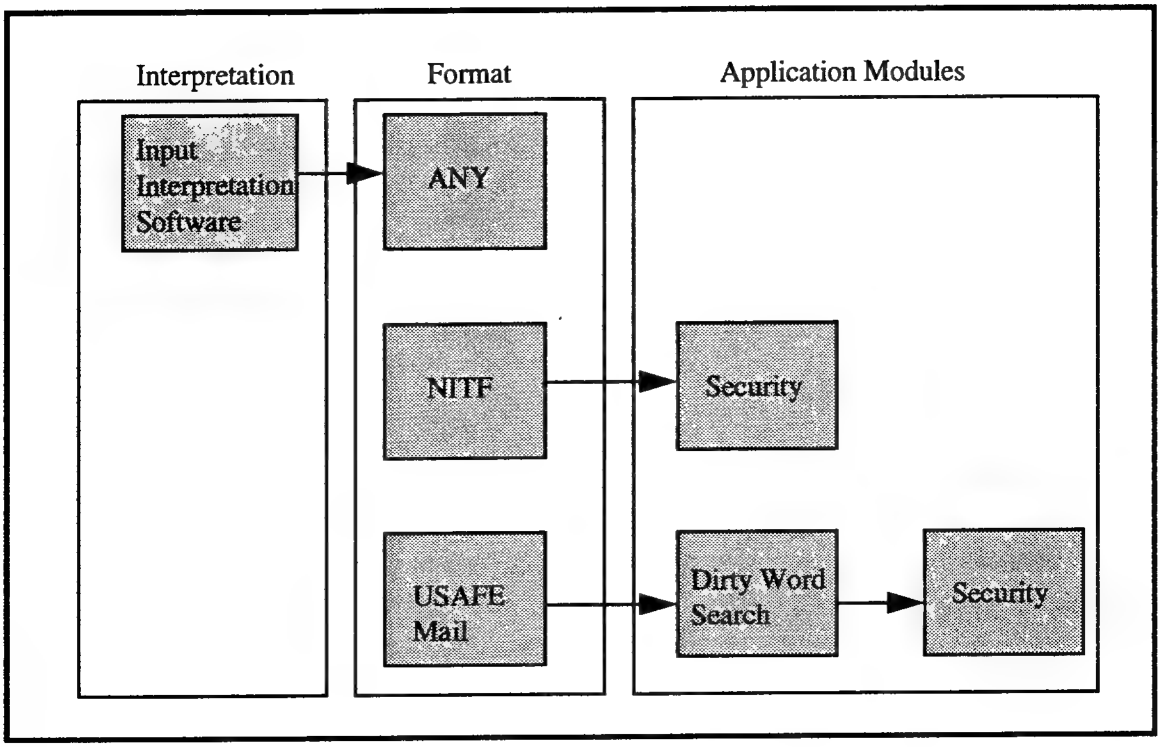


**Figure 5.2.2.2-1 Module Ordering Example**

Release communication modules are not configured as part of the module ordering. This is because they are configured at the guard site and release is performed by finding a matching communications release destination host to send the message on release. It should be noted that if the Dirty Word Search module is before security it will run and use the word list for the input security level. If it is placed after security and security performs a downgrade, the Dirty Word Search module will run at the lower level using the word list at the lower level. It is possible to put a Dirty Word Search module before security and after security to check words for tables at both levels.

### 5.2.2.3 Assembly, Transfer, and Installation

Once the guard node of the configuration or any other node in the system layout has been defined, the Target Guard System is ready to assemble the information for transfer to the *transport media*. Operations on assemble include creating a list of all selected modules and the paths to where the module code exists, creating the Process Configuration Table identifying processes valid on the guard, and creating a checksum of all the information that will be transferred to tape for validation on installation.

Since software selected may exist at more than one level, there may be multiple tapes generated for a single guard environment. It is the responsibility of the operator to attach the appropriate label to the transport media as instructed by the Configuration

Utility. The levels of data for the selected configuration node are displayed to the operator in the transfer window. The operator must pick the transfer device where the TGS will be written along with the level of data. The status of each module transferred is then displayed as it is written to the transport media.

The resultant transport media is then used to install the Target Guard System. Upon install, all software is loaded, tables are converted from text into COTS specific format, and all the modules are compiled and linked. This is all performed using a script prompting the user for appropriate action.

### 5.2.3 Audit Interface

The audit interface is a separate process that allows the operator to review the Configuration Utility audit information. This includes system events, user events, and configuration specific events. This utility is functionally equivalent to the Audit Interface used in the Guard Software as discussed in section 5.3.2.4. The audit events are different in that they are GCU specific instead of Guard specific.

### 5.2.4 System Control

The control process is a trusted process that starts and monitors the execution of all the processes in the GCU. It starts, stops and checks the health of each process and determines when a user logs in. The controlling process software used for the GCU and Guard control process is the same code. When the code is compiled for GCU data product specific processing is not included in the GCU executable. There are also a few minor differences in the functions performed. The functioning of the control process is explained in the Guard section.

### 5.2.5 Audit

The audit process is a trusted process that processes audit event requests from running processes. It receives requests from GMS and CU at various levels to record audit events. Based on the audit event request received, fields of a fixed length audit record are populated and written to a system high audit log. Other requests handled by the audit process include closing the log and starting a new one and responding to status requests from system control. The audit process software used for the GCU and Guard control process is the same code. When the audit process is compiled for the GCU only GCU specific audit events are processed. The functioning of the audit processes is explained in the Guard section.

### 5.2.6 Security

The security process is a trusted process that processes requests from untrusted processes to break operating system security process. It handle requests to add and delete from tables as well as reclassifying data in tables.

## 5.3 GAAP Guard Software CSCI Architecture

The GAAP Guard Software (GGS) consists of interactive administration window software, a core of control, auditing and security software, and a number of guard application modules. Figure 5.3-1 shows how these components fit in a target guard environment. The shaded applications are part of the core software that exists for all generated guards. This figure also shows how all the software is built on top of *Environment Dependent Software* (EDS) to allow portability to the three supported multi-level platforms. The communications, interpretation, and application software is used to pick up and process the data as defined during the configuration process previously described.

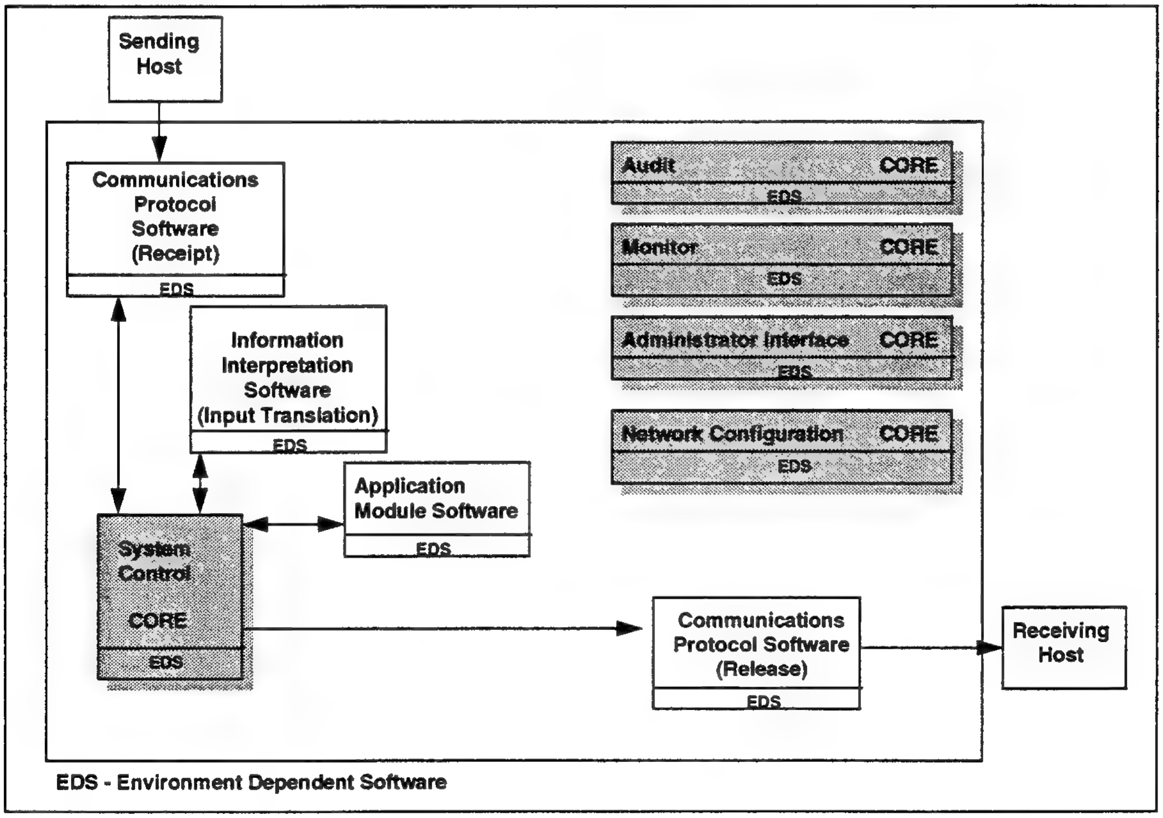

Sending Host
Communications Protocol Software (Receipt)
EDS
Information Interpretation Software (Input Translation)
EDS
Application Module Software
EDS
System Control
CORE
EDS
Audit
CORE
EDS
Monitor
CORE
EDS
Administrator Interface
CORE
EDS
Network Configuration
CORE
EDS
Communications Protocol Software (Release)
EDS
Receiving Host
EDS - Environment Dependent Software

**Figure 5.3-1 GAAP Guard Software**

### 5.3.1 Guard Core Software

The sections that follow will describe the functions performed by the GAAP Guard Core software. First the startup and receipt of messages is described, followed by an example of how messages are processed. Other health check functions are discussed followed by descriptions of the operations of the security and audit processes.

There are several tables used in the Guard Core Software for maintaining information about processes, network communications, routing and error handling. A summary of the tables used and a short description are shown in Table 5.3.1-1.

| Table Name | Description |
|---|---|
| Process Configuration Table | Contains information about configured processes. This table is updated when input and output communications are added. |
| DP Routing Table | Defines the routing information from module ordering. |
| Network Information Table | Defines network information for input and output communications modules |
| Host Information Table | Defines host configuration information |
| Device Table | Defines the valid devices for writing audit archives |
| Audit Log | System high audit log table |
| Audit Criteria Table | Defines events to be audited |
| DP Error Table | Defines processing for error actions as defined in the GCU |
| Port Information Table | Defines the guard system ports |
| GAAP Internal Data Structure (GIDS) Table | Records holding information about data being processed |
| Disk Space Table | Configuration of disk space checking |

**Table 5.3.1-1 GAAP Guard Software Table Descriptions**

### 5.3.1.1 Startup and Message Receipt

The GAAP Core software is used to provide a framework where new modules can be easily introduced and incorporated into pre-established message processing software. Other parts of the framework include guard system auditing and a security process to perform necessary override operations and downgrading. Much of the configurablility for a module is defined in the *Process Configuration Table* (PCT). This information tells the system control process the module application type, the state in which the module is allowed to run, the executable name of the module, authorizations to be passed to the module, and any associated interactive configuration programs. Information contained in this table is shown in Table 5.3.1.1-1.

| Field Name | Description |
|---|---|
| Application ID | Unique id of this application |
| Application Type | Type of Application |
| Executable Name | Name of the executable |
| Processing Mode | States in which the process is allowed to run |
| Displayable Name | Displayable name of the process |
| State After Initialization | State of process after initialization |
| Multi-Level Application | Flag to indicate process is or is not multi-level |

| | |
|---|---|
| Start Level | Starting level of the process |
| Configuration Application ID | Unique id of configuration application |
| User Authorizations | Process authorizations |
| Process Queues | Queue ids required by the process |

**Table 5.3.1.1-1 Process Configuration Table Information**

The system control process uses this information on startup to determine the input communications modules and starts them at the appropriate level. It uses this table for all new processes started to ensure the guard is in the correct state to start the process, determine queues to create for the process, and authorizations to pass to the new process. The startup states of system control are shown in Figure 5.3.1.1-1.

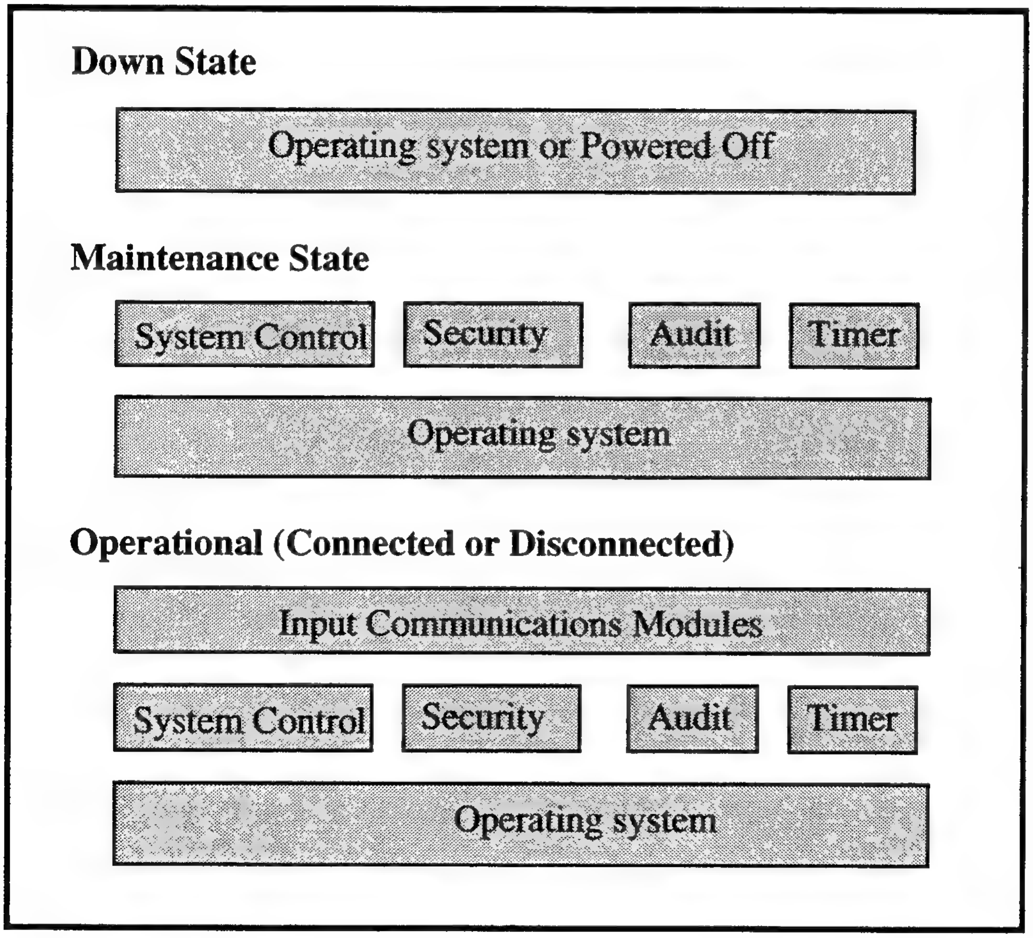


**Figure 5.3.1.1-1 System Control Startup Operations**

In the *down state* either operating system maintenance is being performed or the machine is shut off. When the machine is powered on, system control will automatically be started. It will proceed to start the audit and security processes and wait for positive responses. If there is a problem with starting either of these processes system control will exit. Otherwise, the Process Configuration Table is read to get the names of applications defined to be input communications type. These modules will then be started at the correct level. Upon receiving positive status responses from the input communications modules, they are sent a connect request from system control to complete transition to the *operational connected state*. The guard is then ready to receive messages.

### 5.3.1.2 Message Processing

When an input communications module receives a message, it creates a *GAAP Internal Data Structure* (GIDS) record for the newly received message. The GIDS contains the name and location of where the message to process resides on the guard host and is assigned a unique GIDS id to be used for the life of the message. In our architecture a *data product* is defined to be the message that was received and the GIDS record that describes information about that message. Once an input communications module has notified system control of the new data product, it continues to poll for new messages.

System control will receive the *Data Product (DP) Request* message and begin the process of starting new applications for processing. The Data Product Routing Table is referenced to check the category and step to determine the next action for the DP Request. For new data products it will find the format to be undetermined and recognize the information interpretation module as the first module in the "Any" row of the routing table created during guard configuration. It will start the interpretation module using information in the PCT at the same level of the data product. It will then send a DP Request message to the interpretation module to request processing of the data product.

Using our example module ordering defined in 5.2.2.2-1 lets assume the interpretation finds the data product to be a USAFE Mail type. The interpretation module will verify the source and destination are valid then send a DP Request back to system control to continue its processing. System control will then access the DP Routing Table and will jump the row corresponding to processing a USAFE Mail type message. It will find that this requires starting the dirty word search checker at the level of the data product and will send a DP Request to dirty word search once successfully started. When system control receives a DP Request from dirty word search it will again access the DP Routing table to find the next module. In our example, this would be the security process that performs the downgrade operation prior to release. This module is already running, so system control merely sends the DP Request to the security process. The security process will validate the classification extracted by the interpretation module and will reclassify both the message and the GIDS record to the desired level of release.

The security process will then send a DP Request to system control. System control will find there are no more steps in the Processing row of the routing table for the format identified and will jump the Send row. Once identified the module is ready for release, system control will find a configured output communications module at the level of the destination host that corresponds to the method of receipt and will start the output communications module at the level of the data product which at this point has been downgraded. A DP Request is sent to the release module and the message is released to the specified destination. Figure 5.3.1.2-1 graphically shows the steps to processing the message described above.

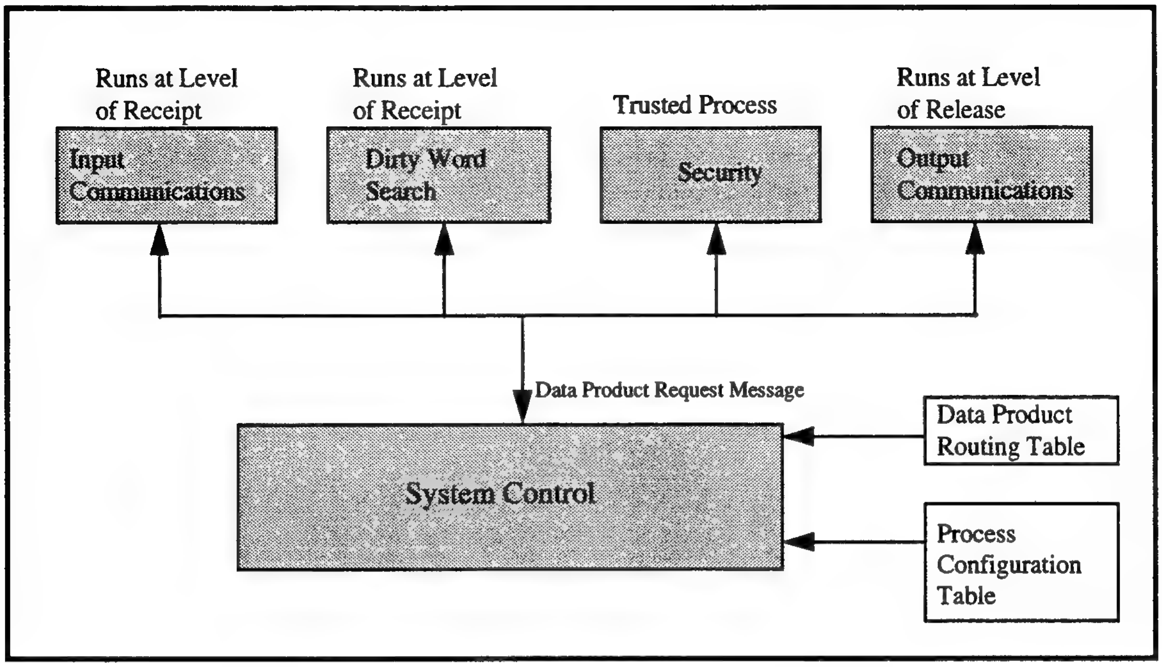


**Figure 5.3.1.2-1 Message Processing Example**

It is important to note that placement of the dirty word search module in the GCU module ordering configuration before or after the security process is significant since the word list it accesses is at the level of the application. If the dirty word search application is before security it will use the input level table and if it is after security it will use the output level table.

### 5.3.1.3 System Health Check Operations

System control and the audit process perform functions related to maintaining the health of the system. The audit process will check the available audit space each time an audit event is received. If the audit space is within 75% a warning is generated. This happens until 90% is reached at which point a warning is generated every 1%. If the audit space becomes 100% full, the system automatically transitions into maintenance state. This is done by the audit process sending a state change request to system control.

System control ensures that the audit process, security process, and other application processes are running correctly by sending status requests to all running processes on a periodic basis. If the processes do not correctly respond, the system transitions to maintenance state. If the audit process or security processes are not running, the guard will automatically shutdown.

System control also checks the amount of disk space being used depending on how it has been configured by the guard administrator. There is a disk space interface program that defines the partitions of the disks to be checked and allows setting the threshold at which the system will go into maintenance mode if disk space is exceeded.

#### 5.3.1.4 Notification

There is a separate process to handle data products marked as having an error condition. Any data product with its error code set is sent to the notification process. The notification process references to DPET configured in the GCU and determines what to do with the DP. Configurable options are to reject the DP, send a notification back to the sending host, or send a confirmation to the sending host that the data product was successfully sent to its destination.

If the particular error has been marked for notification, a new data product is created with the content being a precanned statement followed by any rejection text. The notification data product is then sent to the sending source host via the same method it was received. When a data product has been successfully released, the notification process will send a confirmation message to the sending source if it was configured in the GCU.

The notification process also performs the alternate addressing function. If a data product can not be sent to a destination, the notification process ensures that all alternates for the original host are tried before error processing is done for the data product.

#### 5.3.1.5 Login

The Login process is a process that is invoked when a user logs in. The executable name is in the password file for authorized guard users and is run automatically when a user logs in. It is the only process a guard user id will run directly. The process is run at the level the user logs in at. When the process runs it notifies system control that a user has logged in. System control will then invoke any initial processes the user is authorized to use.

The Login process has a dialog window that is displayed for login error messages and is also used to display visual audit alarms. Visual audit alarm information is displayed in a dialog to the operator for each event configured for visual alarm. The dialog window is not visible unless a visual alarm occurs and can be closed after the alarm is noted.

#### 5.3.1.6 Audit Processing

The audit process is a trusted process that receives requests from interactive and non-interactive software at various levels to record audit events. Based on the audit event request received, fields of a fixed length audit record are populated and written to a system high audit log. Other requests handled by the audit process include closing the log, starting a new audit log, reporting the percentage of audit space available, and responding to status requests from system control.

### 5.3.1.7 Security Process

The security process is a trusted process that processes requests from untrusted processes to break operating system security process. One of the main operations of the security process in the guard is the upgrade and downgrade of data products. The security process will also start the security label interface window to allow the operator to select valid levels when configuring ports, hosts, and communications modules or when a manual classification change is being performed by rejection review. This window varies on all three supported platforms as the operating system's provided calls to use the standard label selection windows. The security label selection window is also used by rejection review to allow upgrade or downgrade of rejected data products.

During network configuration requests are sent to the security process to select the level of a host and write to the Host Information Table. The trusted security process must perform all table write operations to multiple single-level for network configuration runs at system high and is a non-trusted process. When a new communications application is added, security writes to the Network Information Table and the Process Configuration Table. Security writes to the Port Information Table when a port level change is performed.

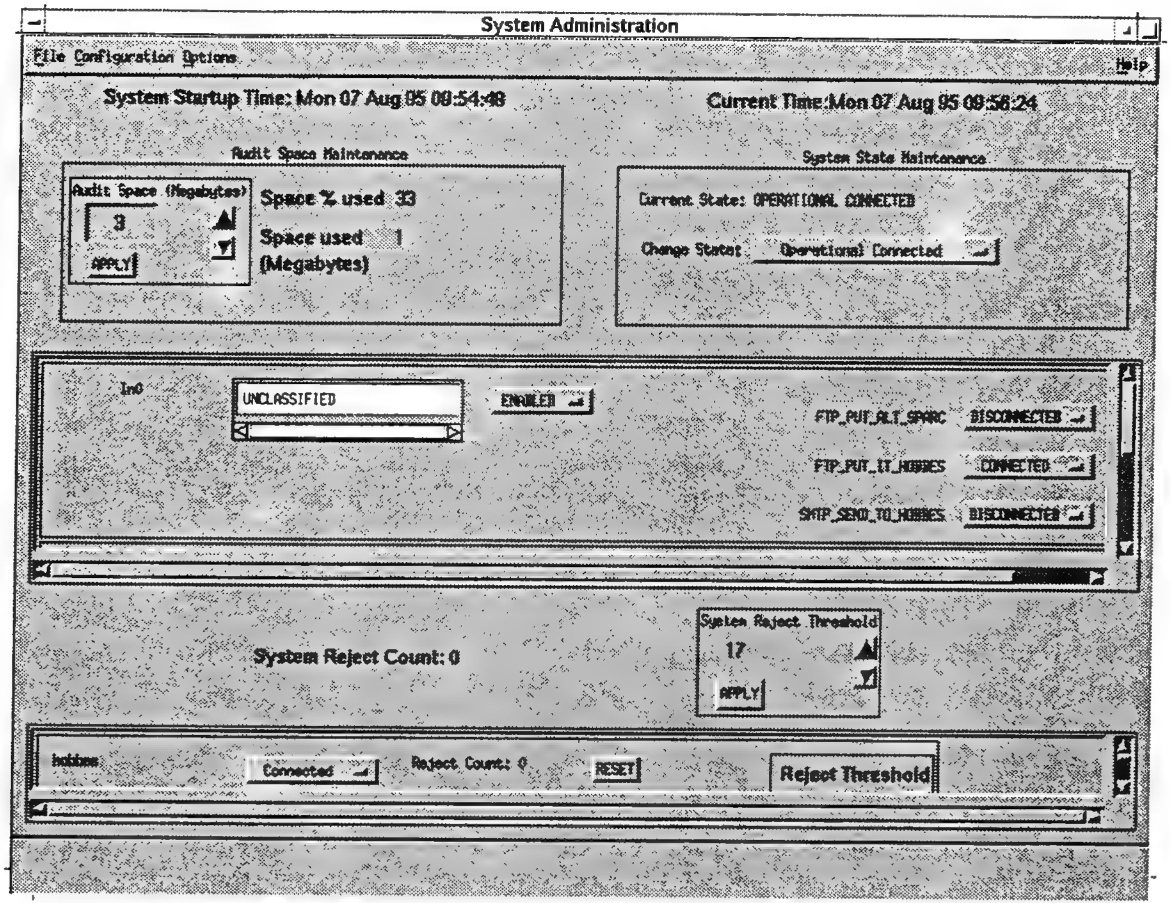
### 5.3.2 Guard Administration

This section will define the guard administrative interface windows that are part of the core software. These interfaces include System Administration, Network Configuration, Monitor, Audit Interface, Audit Criteria, and Disk Space Monitor.

### 5.3.2.1 System Administration

There is an authorization for gaap_admin that allows an operator logged in to see the System Administration interface. System Administration is the base window for administering the guard. It provides capabilities to start new configuration processes, network configuration, monitor, audit interface, and the audit criteria process. It allows for setting and clearing the rejection thresholds, audit space thresholds, and connection and disconnection of ports, hosts, and communications modules. It provides for state maintenance of the guard. The System Administration Window is shown in Figure 5.3.2.1-1.

Based on a users authorization system administration functions may not be available. State change, reject count, audit space and host and port disable functions are available individually if the user is authorized to access the function.



System Administration

System Startup Time: Mon 07 Aug 95 09:54:48

Current Time:Mon 07 Aug 95 09:58:24

Audit Space Maintenance

Audit Space (Megabytes): 3

Space % used 33

Space used (Megabytes) 1

System State Maintenance

Current State: OPERATIONAL CONNECTED

Change State: Operational Connected

In0: UNCLASSIFIED — ENABLED

FTP_PUT_ALT_SPARC: DISCONNECTED

FTP_PUT_IT_HOBBES: CONNECTED

SMTP_SEND_TO_HOBBES: DISCONNECTED

System Reject Count: 0

System Reject Threshold: 17

hobbes: Connected — Reject Count: 0

**Figure 5.3.2.1-1 System Administration Window**

The System Administration window displays the audit space used along with the percentage of audit space allocated. The amount of audit space may be increased through this display. There is an option button used to select the current state of the guard. The possible states and their meaning are defined in Table 5.3.2.1-1.

| State | Meaning |
|---|---|
| Operational Connected | Start all input communications software and send connection message |
| Operational Disconnected | Send disconnect message to all communications software to halt further message processing |
| Down | Take the guard down stopping all processes but leave the operating system running. Allow processes to complete the current function before going down. |
| Immediate Down | Take the guard down stopping all processes but leave the operating system running. Do not wait until processes to complete the current function before going down. |

| | |
|---|---|
| Maintenance | Stop all the application modules leaving all other processes running. |
| Down Including OS | Same as Down but also shutdown the operating system. |
| Immediate Down Including OS | Same as Immediate Down but also shutdown operating system. |

**Table 5.3.2.1-1 Allowable Guard State Changes**

The System Administration window also displays the names of the configured ports with the currently assigned level of the port. For each configured port the communications modules running on that port are displayed showing the name and state of the communications module. There is an ability to enable or disable a selected port or to connect or disconnect a selected communications application. If a port is enabled or disabled, a shutdown and restart of the operating system is required. This is due to operating system constraints that may not exist in newer releases. When a communications module is disconnected it means that process will no longer poll for data. For local receipt types of input via FTP or SMTP mail, that means messages will still be received on the guard host, but they will not be processed. Disabling the port or sending host would be the only way to stop messages from being written to the guard host which currently requires restarting the operating system for both operations.

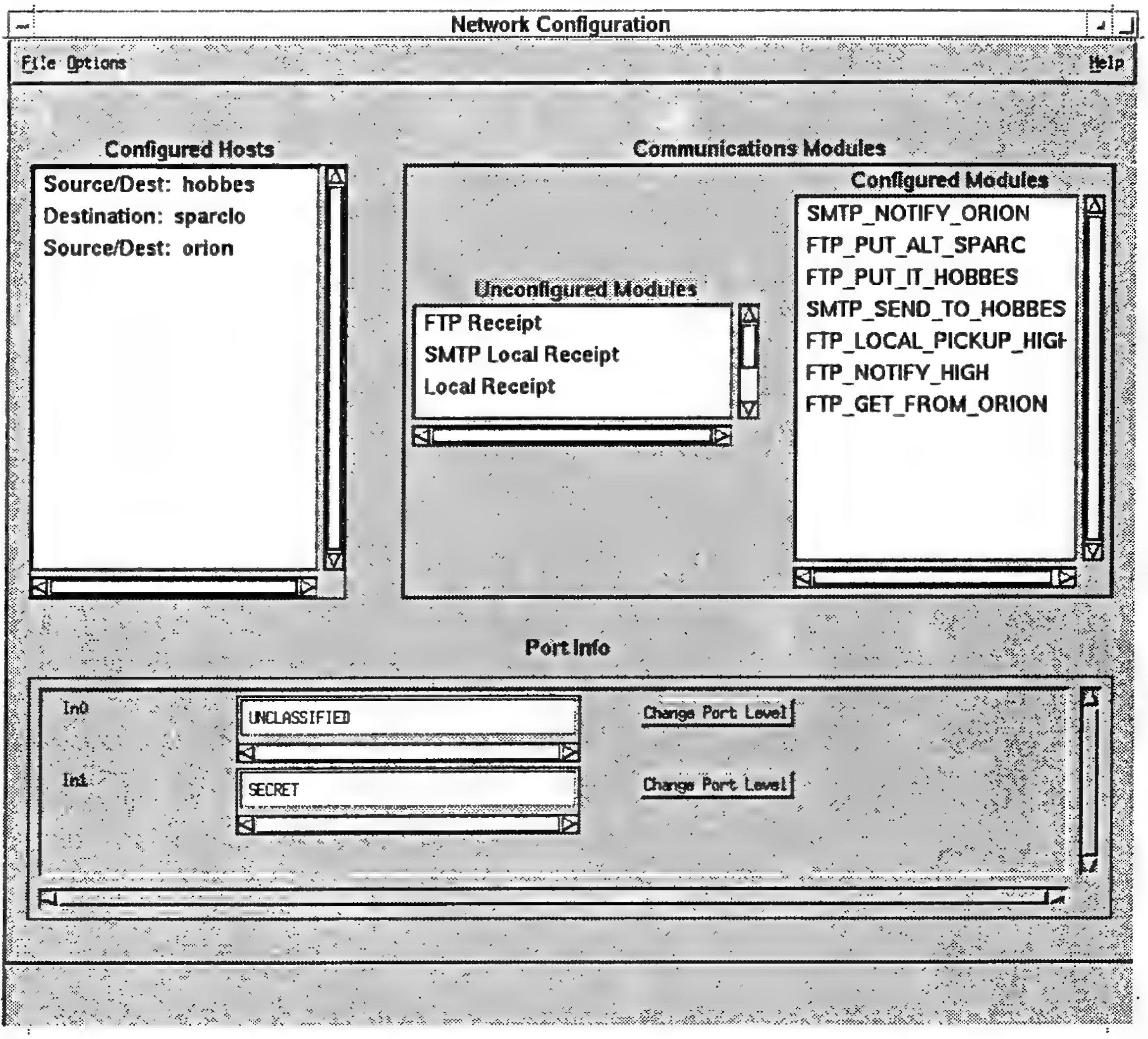
The hosts are displayed showing the configured level of the host. The number of rejected messages are displayed for each host with an ability to increase or decrease the threshold of allowed rejections and clear the current number of rejections. To allow for rejections from an unknown hosts, there is an unconfigured host displayed to tally those. The system rejection count is the sum all the displayed host reject counts and includes an ability to set the total allowable system rejected messages.

Through the menu bar, the operator has the ability to start application specific communications modules, audit interface, audit criteria, network configuration, and the disk space monitor depending on the guard operation role authorizations.

The menu bar on the system administration window is built dynamically. It displays configuration and administration programs options on if the user is authorized to run them. Menu bar options are built based on programs defined in the PCT, so any new configuration or administration programs that may be added in the guard can be displayed as options.

### 5.3.2.2 Network Configuration

In order to run the Network Configuration Window, the operator must be logged in at system high and have the gaap_netconfig authorization. The Network Configuration Window allows for the configuration of valid sending and receiving hosts and configuration of input and output communications modules. It also allows for setting the level of configured ports. The Network Configuration window is shown in Figure 5.3.2.2-1. Network Configuration will only run while in the maintenance state.



Source/Dest: hobbes
Destination: sparclo
Source/Dest: orion

FTP Receipt
SMTP Local Receipt
Local Receipt

SMTP_NOTIFY_ORION
FTP_PUT_ALT_SPARC
FTP_PUT_IT_HOBBES
SMTP_SEND_TO_HOBBES
FTP_LOCAL_PICKUP_HIGH
FTP_NOTIFY_HIGH
FTP_GET_FROM_ORION

In0 UNCLASSIFIED
In1 SECRET

**Figure 5.3.2.2-1 Network Configuration Window**

The first step in configuring a guard system is to define the communicating hosts, ports, and communications software modules. The main display shows a list of the configured hosts, the level of the available ports, and the *configured and unconfigured communications modules.* Configured communications modules are the processes marked in the PCT as Communication In and Communications Out picked in the GCU configuration. They do not run in an operational guard until they have been configured with the necessary communications information.

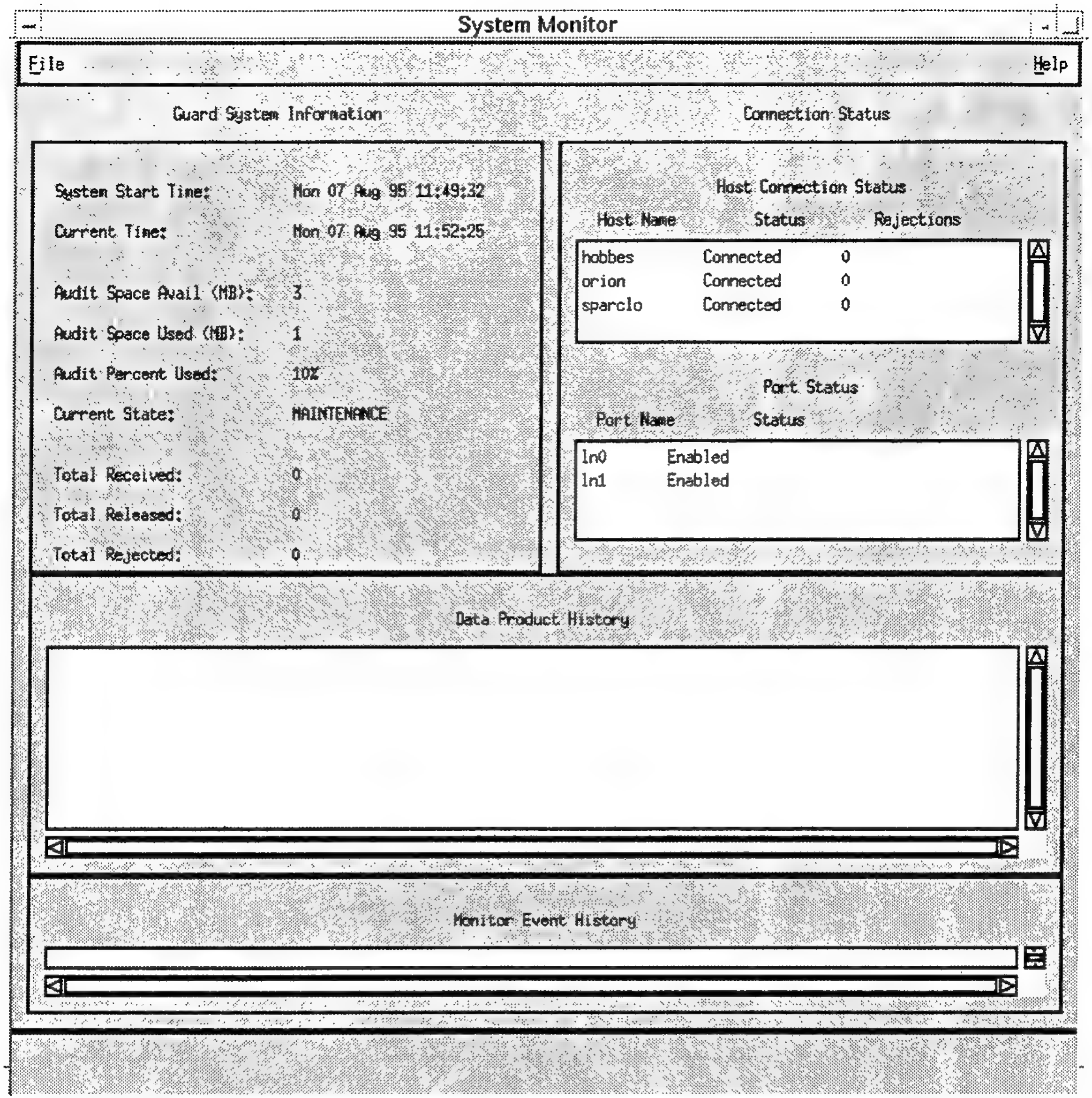
Hosts are configured through the host configuration information window. This window allows for review of configured hosts, adding of new hosts, deleting hosts and alternate address configuration. Alternate addresses define the hosts to be tried in the event the primary host is not available. Available alternate addresses come from the configured hosts list.

The level of a specific port is configured in the Network Configuration window by displaying the platform specific security window for selection of a valid label. This level is used by the operating system to label all data received through that port. Additionally all data transmitted through that port must be labeled at that level. This is also controlled by the operating system. On confirmation, the system files are updated to reflect to new port label. It is very important to realize that the hosts to be communicated with via that port and the communications processes must also manually be configured to run at the same level. It was specifically designed to force the operator to set the level of hosts and have the communications modules marked for the proper level. The current level of the port after the update is reflected on the main Network Configuration window. The port level change is not in effect until the system is rebooted due to constraints on the current versions of the operating systems.

The input and output communications module information is configured through the Network Configuration window. There is a list of the communications modules selected in the GCU. Selecting one of these modules in the list will open a new configuration window for that module. Information typical to communications software includes host names, host addresses, mail addresses, login names, passwords, polling directories, and polling times. Any communication module may have a unique configuration program allowing new modules in the future to be developed independently of the communications methods selected for the GAAP system. The newly configured communications modules appear in the configured modules list and may be modified by selecting the desired module from the list.

### 5.3.2.3 Monitor

The monitor window is displayed when the operator role includes the authorization gaap_monitor. The monitor window shows a picture of the operational guard without allowing or requiring operation interaction. The monitor window is shown in Figure 5.3.2.3-1.



System Start Time: Mon 07 Aug 95 11:49:32

Current Time: Mon 07 Aug 95 11:52:25

Audit Space Avail (MB): 3

Audit Space Used (MB): 1

Audit Percent Used: 10%

Current State: MAINTENANCE

Total Received: 0

Total Released: 0

Total Rejected: 0

Host Connection Status

| Host Name | Status | Rejections |
|---|---|---|
| hobbes | Connected | 0 |
| orion | Connected | 0 |
| sparclo | Connected | 0 |

Port Status

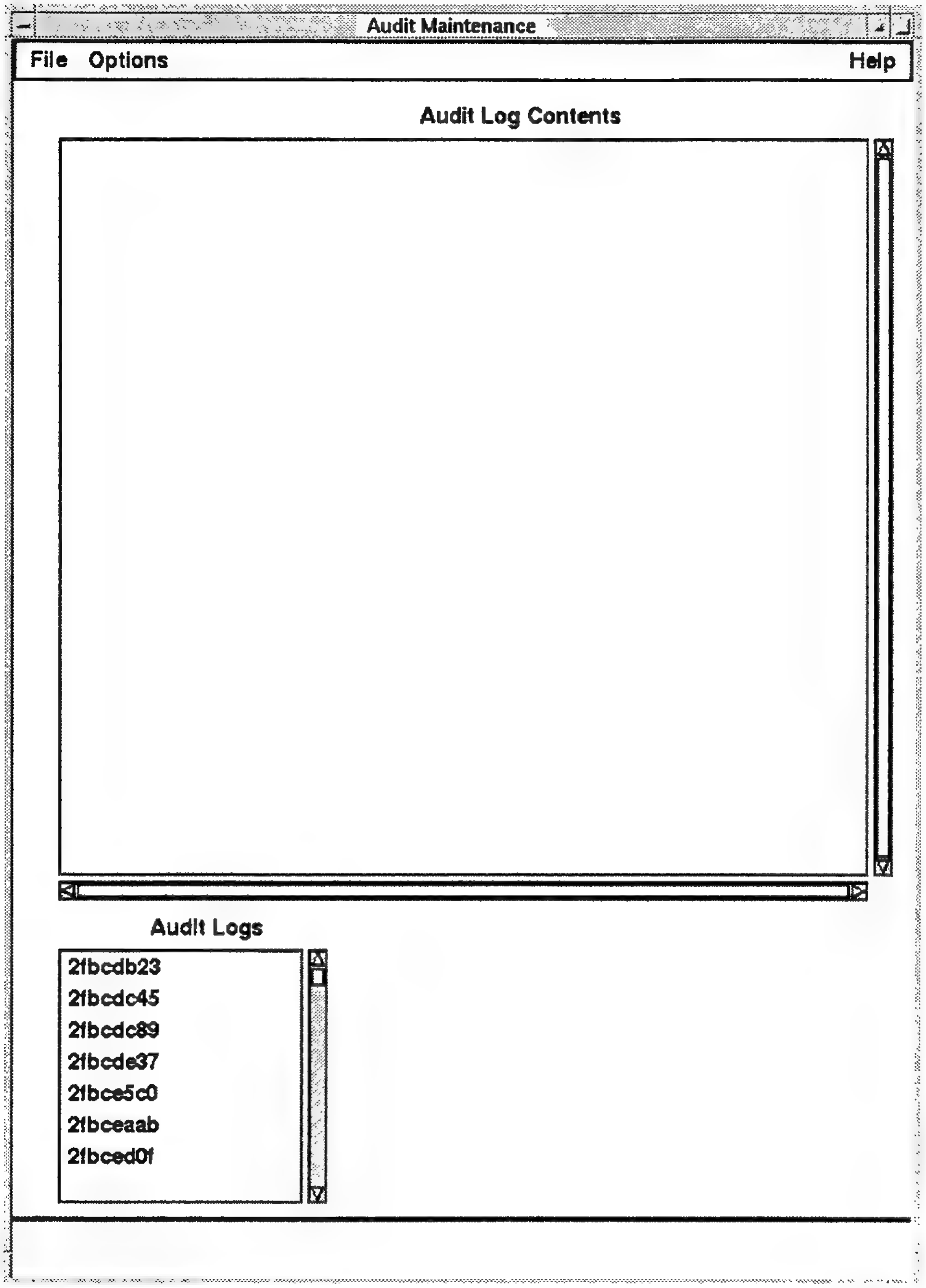
| Port Name | Status |
|---|---|
| ln0 | Enabled |
| ln1 | Enabled |

**Figure 5.3.2.3-1 Monitor Window**

The monitor window shows the current time and the time the system was started. It shows the audit space used and percentage available. The monitor shows a list of the configured hosts and the number of rejected data products from each host. There are counts showing the total number of data products received, rejected, and released. The history of modules that a data product was processed by is displayed upon a data product being released or rejected. This feature quickly identifies how each data product was processed. Finally there is an audit event history that shows each audit event that has been configured to be displayed on the monitor.

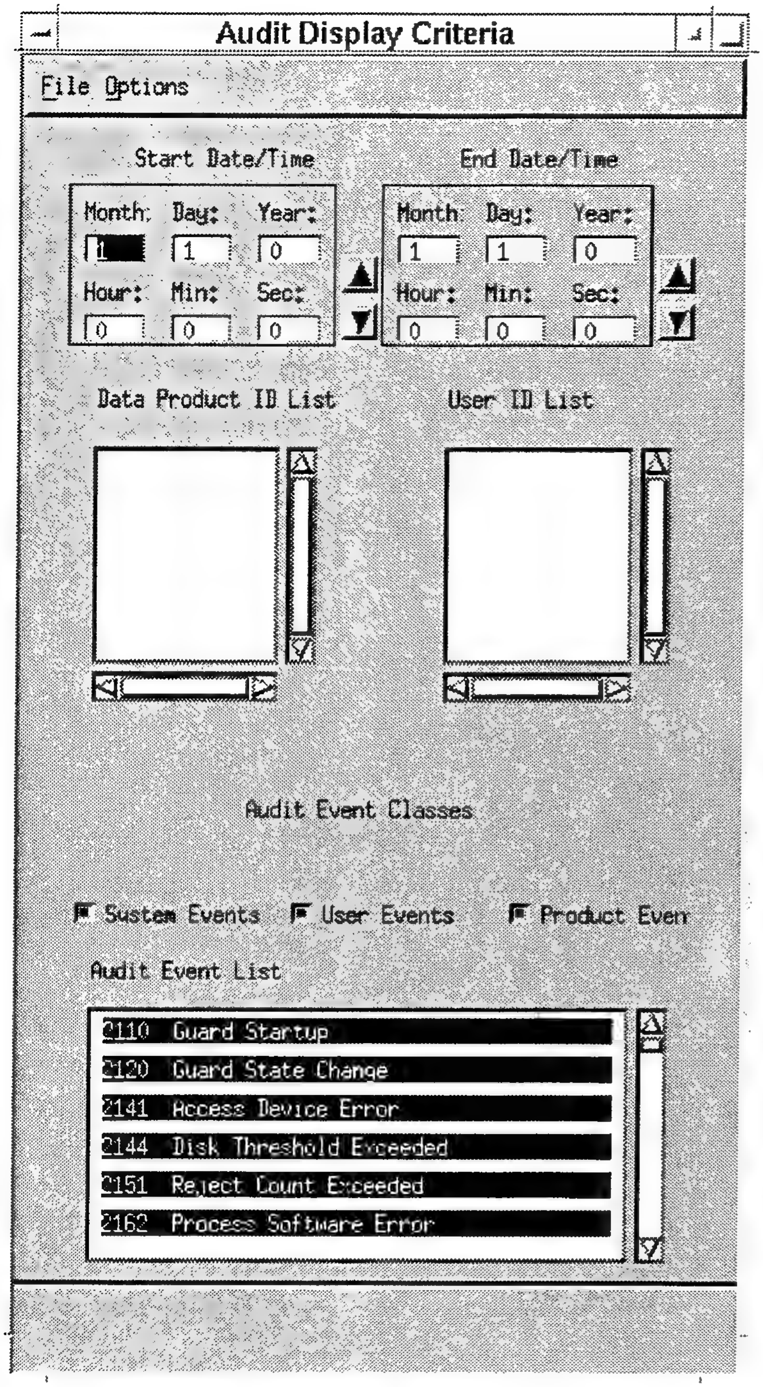
### 5.3.2.4 Audit Interface

The audit interface window is available for display when the operator logged in has the audit review authorization. The audit interface window allows the operator to view, archive, restore, and delete audit logs. When an audit log is being reviewed, there is an audit reduction display used to quickly identify the audit events of interest. The audit review window is shown in Figure 5.3.2.4-1 and the audit reduction window is shown in Figure 5.3.2.4-2.



2fbcdb23
2fbcdc45
2fbcdc89
2fbcde37
2fbce5c0
2fbceaab
2fbced0f

Figure 5.3.2.4-1 Audit Review



Start Date/Time

Month: 1 Day: 1 Year: 0

Hour: 0 Min: 0 Sec: 0

End Date/Time

Month: 1 Day: 1 Year: 0

Hour: 0 Min: 0 Sec: 0

Data Product ID List

User ID List

Audit Event Classes

☒ System Events ☒ User Events ☒ Product Even

Audit Event List

2110 Guard Startup
2120 Guard State Change
2141 Access Device Error
2144 Disk Threshold Exceeded
2151 Reject Count Exceeded
2162 Process Software Error

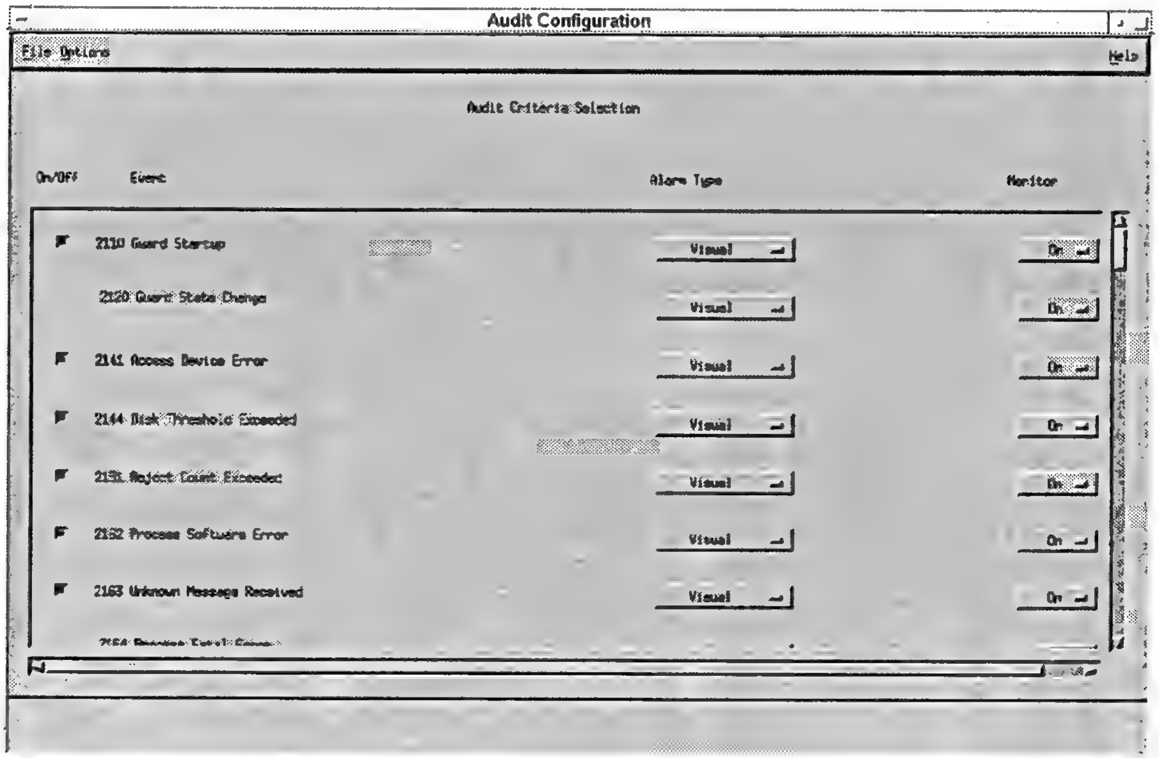
**Figure 5.3.2.4-2 Audit Reduction**

The audit review window displays the available logs with the most recent log at the top. There is a scrolled text window for displaying the audit log event data. Options available on the audit review window include archiving selected logs, restoring selected logs previously archived, and deleting audit logs. There is an option to close the current log thereby causing the audit process to create a new log. The audit log view is a static view of the selected log and is not updated with current events. The current log is not available for deletion.

The audit reduction window is used to locate audit events of specific interest. Filters available include start and end dates, selected data products by ID, selected user events, and options to turn on or off all data product events, user events or system events. There are options to quickly pick all events or deselect all events for the data product, user, and system categories. Select all for the dates is used to return the original start and end dates of the log.

### 5.3.2.5 Audit Criteria

The audit criteria window is used to select the events to be audited, events to be displayed on the system monitor, and events to receive audible and/or visual alarms. The audit criteria window is shown in Figure 5.3.2.5-1.



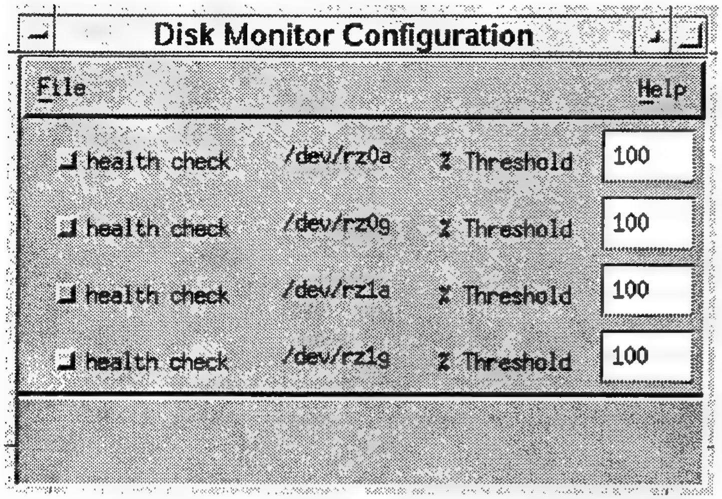
| On/Off | Event | Alarm Type | Monitor |
|---|---|---|---|
| ☑ | 2110 Guard Startup | Visual | On |
|  | 2120 Guard State Change | Visual | On |
| ☑ | 2141 Access Device Error | Visual | On |
| ☑ | 2144 Disk Threshold Exceeded | Visual | On |
| ☑ | 2131 Reject Count Exceeded | Visual | On |
| ☑ | 2152 Process Software Error | Visual | On |
| ☑ | 2163 Unknown Message Received | Visual | On |

**Figure 5.3.2.5-1 Audit Criteria Display**

All events are displayed in this window in numerical order for each available audit event. There is a toggle button to select the event to be audited. If the toggle button is not displayed, that audit event is a mandatory event and is always audited. There are buttons to indicate if the event is to be displayed on the monitor and a selection option that shows that the type of alarm to occur. The choices for alarms are none, audible, visual, or both audible and visual. An audible alarm will cause the terminal to beep and the visual alarm will be displayed in the login dialog box.

### 5.3.2.6 Disk Space Monitor

The disk space monitor is used to configure the disk partitions to be checked by system control. This window is shown in Figure 5.3.2.6-1. This window shows a list of all disk partitions with option buttons for selection to be checked as part of the system control health check. There is also a threshold of percentage full configured to cause a maintenance state change.



| | | | |
|---|---|---|---|
| health check | /dev/rz0a | % Threshold | 100 |
| health check | /dev/rz0g | % Threshold | 100 |
| health check | /dev/rz1a | % Threshold | 100 |
| health check | /dev/rz1g | % Threshold | 100 |

**Figure 5.3.2.6-1 Disk Space Monitor**

## 5.3.3 Guard Application Modules

This section will define the guard application modules used to show processing and filtering messages sent through the guard.

### 5.3.3.1 Communications

Communications modules include receipt and release of data products using FTP and SMTP mail. For each of the communications modules there is a corresponding configuration module that is executed during network configuration to get specific information such as login names, passwords, mail addresses, polling directories, and polling times.

There is a module that will remotely login to a remote host and retrieve files from a specified directory on that remote host. The configuration program for this module gets the remote host name, the login and password for the FTP login, the directory on the remote host, and the polling time to check for files. The files to be retrieved may also be specified with wild cards such as *.dat files.

Files can also be processed when sent from a remote host to a specific location on the guard. For messages received via mail the directory to look for files is pre-configured. The delivery directory may be specified for messages delivered to the guard via FTP. Other information needed for locally received messages includes the polling time, and port name.

Two methods of delivery included in the guard software are FTP and SMTP mail. Releasing a data product via FTP requires a configuration program to get the name of the destination host, login and password information, and a delivery directory. To send via mail the user name is required.

### 5.3.3.2 Interpretation

The interpretation module will recognize the formats USAFE mail, USAFE Tum, NITF, and a newly defined GAAP header. The message source, destination, and classification are extracted from the headers. The source and destination hosts are validated against configured hosts and the message is failed if they do not match. The NITF message must include a header first. The interpretation records the type of message found in the GIDS along with the source, destination, and classification extracted.

### 5.3.3.3 Filtering

Two modules were written to aid in filtering of messages prior to release. The dirty word checker is used to identify words that should not appear in a message to be released. This module has the word checker that validates the words while processing a data product and also has a configuration program used to define the word lists. The configuration program runs at the level of the logged in operator and the word list created is only for that level. The placement of the word search module is thus very critical as to which word list will be used in processing. It is possible to configure a word search module before and after the downgrade operation thereby checking words in the list at the input level and the output level.

The second module written for filtering is a rejection review. This window is available when logged in as the system administrator with rejection review authorization. A list appears of rejected data products for errors that were marked with a reject action when configured in the GCU. When a data product is selected, the content of the message is displayed. If the operator is at the same level as the data being viewed, it is possible to edit the rejected message and perform an action such as forward, resubmit, or reject. Rejecting the data product will cause it to be deleted from the system. Resubmitted data products are sent to the interpretation module and processed through all configured modules from that point. A forwarded data product is released without any subsequent processing and requires the forward on authority authorization.

## 5.4 Configurability

This section will address areas of the Guard system that are configurable. This is done in a separate section from the detailed architecture discussion to keep a focus on all the configurable parts drawing from the presented GCU and Guard architecture. Table 5.4-1 details the areas of the Guard system that are configured either in the GCU or at the Guard site.

| Configurable | Description of Configurability |
|---|---|
| Modules | Any new modules may be added to the GCU. These new modules may be for communications, interpretation, or any type of message processing necessary. |
| Configuration Applications | If a new module requires interactive configuration, a configuration module may be specified. The information exchanged between the configuration program and the guard module can be specified any way the developer desires. |
| Core Modules | It is possible to write new core modules to replace existing core capabilities. A new monitor program or system administration could be added if desirable. |
| Module Order | The order is which a particular message of a specified format is processed is configurable in the GCU. This allows for different type of processing for different types of data. |
| System Roles | The login roles in the TGS are configurable through the GCU to define user authorizations. |
| Authorizations | Authorizations available for configured roles are configured in the GCU. |
| Error Processing | The notification action to take on error conditions while processing are configured in the GCU. |
| Module Information | Information about modules is defined in the GCU for reference. |
| Network Information | Information about the network setup, classification levels, and communications processes are configurable on the guard. |
| Host Information | Host specific information is configurable on the guard.. |
| Disk space check | Disk partitions to be checked for selected thresholds are configured on the guard. |
| Audit Threshold | The audit space threshold is configured on the guard. |
| System rejection count | The total allowable system rejection count is configurable at the guard. |
| Host rejection count | The total allowable number of rejections for a specific host is configurable at the guard. |

| Configurable | Description of Configurability |
|---|---|
| Port Levels | The level of the communication ports are configurable on the guard. |
| Host Levels | The level of each specific host is configurable on the guard. |
| Communication Module Levels | The level of each specific communications module is configurable on the guard. |
| Audit Criteria | The audit events to be audited are configurable on the guard. |
| Audit Events | Audit events can easily be defined and added in the implemented architecture |
| Audit Alarms | Audible and visual alarms are configurable |
| Allowable Run State | The allowable state in which a module is run is configurable. |
| Process Queues | Access to the security process queue is configurable. |
| State to run | The allowable state that a module can run is configurable. |

**Table 5.4-1 Configuration Options**

## 6. Final Testing Results

Preliminary testing of the GAAP System was performed in August of 1995. During this testing, the Configuration Utility Procedures were performed and most of the Guard Software functions were demonstrated. The desired network setup was identified during the Preliminary testing to be used in formal testing at Rome Laboratory. Formal testing of the Guard Software was performed in September of 1995 at Rome Laboratory using a network setup shown in Figure 6-1.

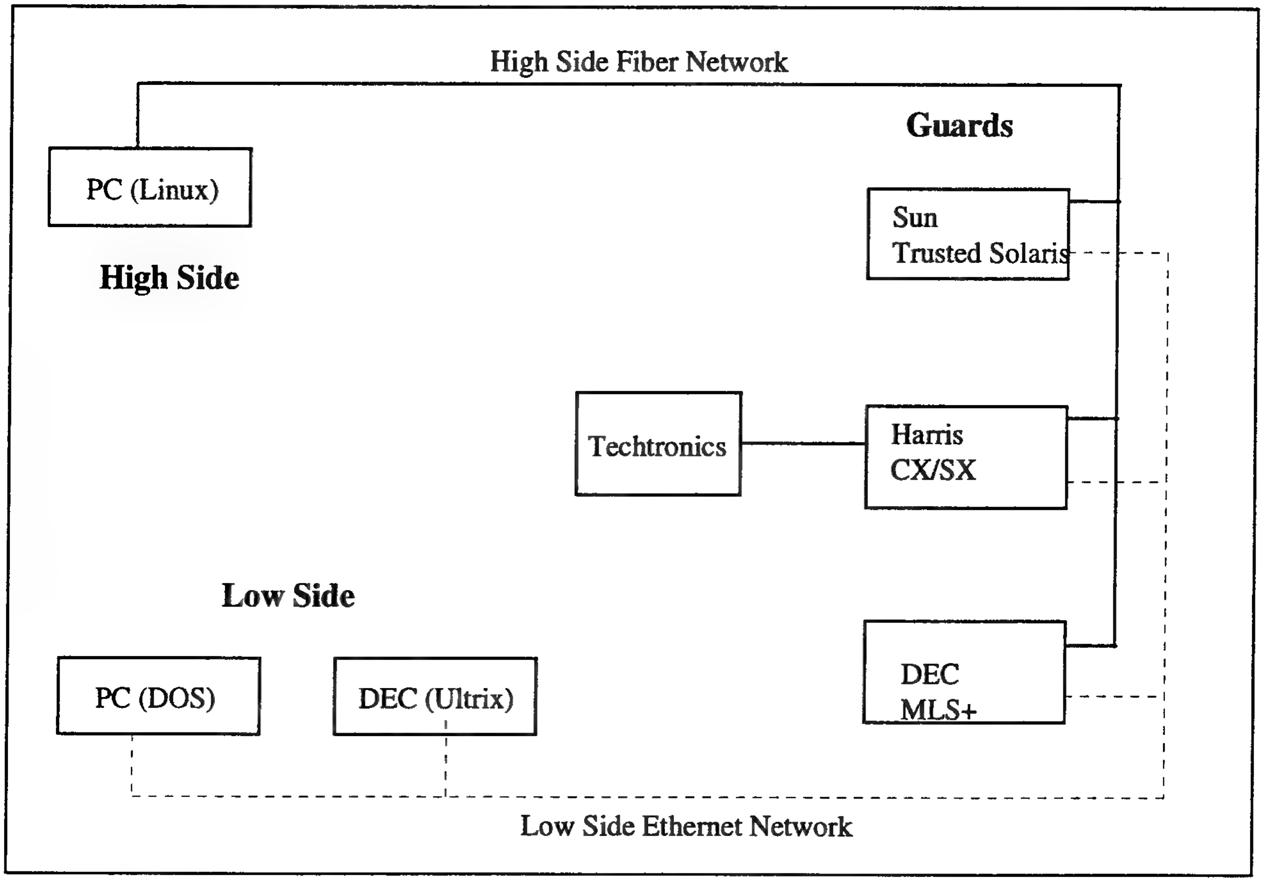


**Figure 6-1 Rome Laboratory Formal Testing Environment**

During a one week period the network configuration and setup was performed and Guard Software was generated from the Configuration Utility to run on the Sun, Dec, and Harris platforms. The test procedures were formally witnessed and system level testing was performed while demonstrating the capabilities of the GAAP Guard Software.

In November of 1995 the final software delivery was made to Rome Laboratory along with all final documentation. During the week in Rome acceptance procedures were run on the Configuration Utility and on the GAAP Guard Software procedures. Procedures were run on the Configuration Utility as assurance that any changes made did not affect requirements satisfaction as a number of enhancements had been added since that time. System testing was also performed on the Configuration Utility in addition to running the test procedures. Figure 6-2 shows the final results of the GAAP Configuration Utility tests and Figure 6-3 shows the final results of the GAAP Guard Software tests.

| Target Guard System Test | Results |
|---|---|
| 1.1 - Hardware Configuration | Success |
| 1.2 - Software Configuration | Success |
| 1.3 - Build and Install | Success |
| 1.4 - GAAP Maintenance | Success |

**Figure 6-2 GAAP Configuration Utility Final Test Results**

| Target Guard System Test | Results |
|---|---|
| 2.1 - Network Configuration | H - Success<br>S - Success<br>D - Success |
| 2.2 - Administration and Monitor | H - Success<br>S - Success<br>D - Success |
| 2.3 - Audit | H - Success<br>S - Success<br>D - Success |
| 2.4 - Communications | H - Success<br>S - Success<br>D - Success |
| 2.5 -<br>Information Interpretation Software | H - Success<br>S - Success<br>D - Success |
| 2.6 - Application Module Software | H - Success<br>S - Success<br>D - Success |
| 2.7 - Spare | NA |
| 2.8 - Spare | NA |
| 2.9 - User Interface | H - Success<br>S - Success<br>D - Success |

Results for indicated platform: H = Harris S = Sun D = DEC

**Figure 6-3 GAAP Guard Software Final Test Results**

As the final test results indicate, the final software delivery satisfied all the identified software requirements and system testing was performed to demonstrate the abilities of both the Configuration Utility and the GAAP Guard Software as developed for the GAAP System. All final documentation was completed and delivered to accompany the hardware and software at Rome Laboratory.

## 7. Recommendations

The GAAP system was developed for Rome Laboratory by Fuentez Systems Concepts during the period of August 1993 to November 1995 as part of a 27 month effort to develop a system that allows for rapid construction of guard systems that are easily portable to multi-level secure platforms. The concept of this project was created on the basis that currently developed guard systems are too functionally specific or too platform specific and that substantial cost savings could be utilized by developing a framework that would allow for both portability and adaptability to new guard requirements in a cost effective manner.

It was identified that the system should be highly configurable to allow for the addition of new modules and flexible to adapt to the operational requirements of almost any guard environment. It was proposed that one CSCI would be a Configuration Utility used to maintain all the modules, provide for auditing and backup of the configuration system, allow for the construction of the operational guard environment, and creation of the target guard system itself. The second proposed CSCI was the Guard Software that would establish the core framework for all generated guards and develop a small sample set of applications to demonstrate the guard capabilities. These sample applications included communications, translation, and filtering.

This paper has outlined the conception and implementation of the GAAP System defining a system that is portable to three multi-level platforms and that is highly configurable to allow for easy modification for almost any operational scenario. The fact that it is so configurable has made it hard to classify the application domain for use of the GAAP System. It can be used in any environment where there is input, processing, and output with a need to control the processing order, audit the system events, configure the network communications, and monitor daily system operations. Table 7-1 shows a few examples of where this technology could be applied.

| Application Domain | Example Usage |
|---|---|
| High to Low Guard<br>Low to High Guard | Current developed modules can be used to accept data from a defined host either at a High level or a Low level and allow filtering and downgrading of information processed. |
| Review / Release | The rejection review application developed for the GAAP System allows for review and release of rejected messages. By developing one new module, this could be expanded to allow review of all data prior to release or use specially developed filters for reviewing based on areas of interest and/or sources of receipt and release. |
| Sanitization | A sanitization or normalization module could be developed and inserted to automatically modify specific message content. |
| MISSI Trusted E-Mail | MISSI compliant communications modules could be |

| | |
|---|---|
| | developed for receipt and release of E-Mail using Fortezza technology. |
| Firewall | The GAAP System could be used to setup restrictions on communicating hosts and communications methods to act as a firewall between any two networks. |
| LAN / WAN Configuration | The Configuration Utility is generic enough to be used to define any networking environment allowing for definition of hardware and software for nodes on a LAN or WAN. |

**Table 7-1 Example uses of the GAAP System Technology**

The GAAP System as implemented provides the method for configuration of guard systems and the implementation of a core of software that is portable and flexible for meeting the needs of guard type systems.

## 8. References

| DOCUMENT ID | TITLE |
|---|---|
| AFR-800-14 | Life-Cycle Management of Computer Resources and Systems, 29 September 1986 |
| NCSC-TG-005 | Version-1, Trusted Network Interpretation (TNI), 31 July 1987. |
| NCSC-TG-006 | Version-1, A Guide to Understanding Configuration Management in Trusted Systems, 28 March 1988. |
| DIA | DODIIS Reference Model for the 1990's, 18 Oct 1991. |
| DIAM 50-4 | Security of Compartmented Computer Operations (U), Confidential, 24 June 1980. |
| DIAM 50-4 | DOD Intelligence Information Systems (DODIIS) Computer Security (COMPUSEC) Program, Preliminary Draft, November 1992. |
| FIPS PUB 158 | The User Interface Component of the Applications Portability Profile, National Institute of Standards and Technology, 29 May 1990. |
| MITRE.03 | GAAP System Security Requirements, Concept Phase: Draft December 1993. |
| MITRE.04 | GAAP Concept of Operations, Concept Phase : Draft September 30, 1993. |
| GAAPSSS01 | System/Segment Specification for the Guard Architecture for Application Portability (GAAP) Final, Fuentez Systems Concepts, Inc. (FSC), 10 November 1995. |
| GAAPSSDD01 | System/Segment Design Document for the Guard Architecture for Application Portability (GAAP) Final, Fuentez Systems Concepts, Inc. (FSC), 10 November 1995. |

GAAPSRS01 Software Requirements Specification for the Guard Architecture for Application Portability (GAAP) Draft for the GAAP Configuration Utility Software CSCI, Fuentez Systems Concepts, Inc. (FSC), 10 November 1995.

GAAPSDD01 Software Design Document for the Guard Architecture for Application Portability (GAAP) Draft for the GAAP Configuration Utility CSCI, Fuentez Systems Concepts, Inc. (FSC), 10 November 1995.

## 9. Acronyms and Abbreviations

This section provides definitions for all acronyms and abbreviations that appear in this document.

| | |
|---|---|
| AMS | Application Module Software |
| CM | Configuration Managed |
| CPS | Communication Processing Software |
| CRC | Cyclic Redundancy Check |
| CSCI | Computer Software Configuration Item |
| CU | Configuration Utility |
| DP | Data Product |
| EDS | Environment Dependent Software |
| FSC | Fuentez Systems Concepts, Inc. |
| FQT | Factory Qualification Test |
| GAAP | Guard Architecture for Application Portability |
| GCS | GAAP Core Software |
| GCU | GAAP Configuration Utility |
| GGS | GAAP Guard Software |
| GIDS | GAAP Internal Data Structure |
| GMS | GAAP Maintenance Software |
| IIS | Information Interpretation Software |
| NITF | National Imagery Transmission Format |
| NSA | National Security Agency |
| LAN | Local Area Network |
| MLS | Multi-Level System |
| RL | Rome Laboratory |
| SRS | Software Requirements Specification |
| STP | Software Test Plan |
| TCSEC | Trusted Computer System Evaluation Criteria |
| TGS | Target Guard Systems |
| TUM | Transaction Update Message |
| USAF | United States Air Force |
| USAFE | United States Air Force Europe |
| WAN | Wide Area Network |

## 10. Glossary of Terms

### Accreditation

A formal declaration by DAA that the Automated Information System is approved to operate in a particular security mode using a prescribed set of safeguards.

### Certified

The designation by the system operator that a module has been tested to run on the designated platform. It in no way implies that the module has passed any particular security testing.

### Configuration Utility (CU)

The software in the GCU that allows configuration of the hardware and software used to build the Target Guard System (TGS).

### Classes

Statically defined high level categories used to load new modules. Classes contain lower level groups of modules.

### Configured and Unconfigured Communications Modules

A configured communications module is one that will be started by system control in the operational state and has the required configuration information necessary to receive new files.

An unconfigured communications module is one that has been selected for the guard during the configuration process and is available for selection to be configured. Unconfigured communications modules are never started by system control.

### Core Software

A set of guard software that performs control, auditing, security, and administration and is part of all generated guard systems.

### Data Product (DP)

A Data Product is the name applied to a file that is being processed and the added information contained in the GIDS record that defines the processing. Each item processed by the guard is described as a data product rather than a file.

**Data Product Request**

A message used by all modules to communicate the processing of a data product. Modules will receive a Data Product Request when new data is to be processed. Modules will send a Data Product Request to system control when the module has completed processing the data product.

**Down State**

A guard system state in which no GAAP system software is running.

**Environment Dependent Software (EDS)**

A low level of software that ensures higher level code with run on multiple platforms without modification.

**GAAP Configuration Utility (GCU)**

The GCU is one of the two main CSCIs in GAAP. It allows for loading and maintaining guard modules and for the configuration of the operational guard environment.

**GAAP Guard Software (GGS)**

The GGS is the collection of software that is loaded from a Target Guard System (TGS). It includes the core software for control and administration along with any specifically selected modules used for processing data products.

**GAAP Header**

A GAAP system specific defined header that contains minimal required information such as source, destination, and classification.

**GAAP Internal Data Structure (GIDS)**

The GIDS record defines the information known about a particular file being processed by the guard. It is accessible by all guard module software through a set of common calls.

**GAAP Maintenance Software (GMS)**

The software in the GCU that is used to load and maintain guard modules. It also includes the administrative abilities for audit review and backups.

**Groups**

A sub-level of categories within the statically defined classes used to further distinguish different types of modules that have been loaded into the GCU.

**Level**

The combination of a hierarchical classification and a set of non-hierarchical categories that represent the sensitivity of information.

**Message**

A message is a notification of an event such as a rejection notification.

**Module**

A module is a set of software that follows the GAAP ICD to be incorporated in the GAAP system. It is maintained by system control and defined to start and stop as defined by the module summary information.

**Multi-level System**

A class of operating systems that segregate data at specified security levels and provide assurance for DAC and MAC access to data at various levels.

**NITF Header**

Header format specific to NITF binary messages.

**Node**

An object in the configuration utility system layout that could be a workstation, PC, WAN, LAN, Router, or any other type of device connected to a network.

**Operational Connected State**

The guard system state in which security, audit, and control are running and the input communications modules are receiving new input data.

**Product**

Products are software components being added to the GAAP baseline or being selected for a TGS.

**Process Configuration Table (PCT)**

The PCT is a table that defines all required information for the system control process to determine when and how to start modules.

**Target Guard System (TGS)**

The TGS is the guard system that was built and created through the configuration utility. It includes the selected software and specific tables defining the module ordering, error handling, and authorization information. The TGS is installed at the guard site to create an operational guard system.

**Transport Devices**

Devices that can be used in transport of data to include for example 4mm tape drives or Cartridge tape drives.

**Transport Media**

The physical container, such as 4MM DAT tape, used for adding modules to the baseline or transporting products to the TGS.

**USAFE Tum Header**

A specific USAFE header format used for sending messages via FTP.

**USAFE Mail Header**

A specific USAFE header format used for sending messages via SMTP.

*U.S. GOVERNMENT PRINTING OFFICE: 2001-610-055-10123

## *MISSION*
## *OF*
## *AFRL/INFORMATION DIRECTORATE (IF)*

*The advancement and application of Information Systems Science and Technology to meet Air Force unique requirements for Information Dominance and its transition to aerospace systems to meet Air Force needs.*